

PENGEMBANGAN RANCANG BANGUN GAME EDUKASI LOGISTIK "STOWAGAME" MENGENAI PENATAAN KONTAINER DI BAY KAPAL

HENDRIYONO RACHMAN, DAN AHMAD RUSDIANSYAH

Jurusan Teknik Industri, Institut Teknologi Sepuluh Nopember (ITS) Surabaya

E-mail: arusdian@ie.its.ac.id

ABSTRAK

Pengembangan game edukasi yang dibuat fokus pada permasalahan yang terjadi pada manajemen pelabuhan utamanya pada permasalahan loading kontainer di suatu bay kapal. Permasalahan dalam perencanaan loading di bay kapal biasa disebut dengan Master Bay Plan Problem (MBPP). Kompleksitas dari permasalahan MBPP dapat dilihat dari konstrain yang ada, yaitu berat kontainer, tujuan kontainer, ukuran kontainer, tipe kontainer, dan aturan peletakan kontainer. Dalam pembuatan game, studi kasus, aturan permainan dalam pengembangan game ini disesuaikan dengan konteks teori dan juga kondisi nyata yang terjadi pada penataan kontainer di bay kapal. Rancang bangun dari game ini dinamakan StowaGame. StowaGame adalah game edukasi berbasis simulasi terkait dengan penataan kontainer di bay kapal. Perangkat lunak yang dibuat menerjemahkan pengetahuan dari MBPP ditambah unsur hiburan ke dalam aturan game. Untuk menguji keefektifan game edukasi ini, telah dilakukan uji kepada para pemain. Pengetahuan mengenai MBPP dari tester akan diuji sebelum dan sesudah memainkan game. Perhitungan uji paired-t test menghasilkan p-value < 0.05 sehingga dapat disimpulkan bahwa game edukasi yang dibuat mampu memberikan edukasi yang signifikan terkait dengan penataan kontainer di bay kapal.

Kata kunci: game edukasi, master bay plan problem, penataan kontainer

ABSTRACT

Development of educational games focuss on the problem occured on bay management, primarily on loading container at ship bay. The loading plan problem in ship bay commonly known as Master Bay Plan Problem (MBPP). The complexity of MBPP can be seen on its constraints, that are container weight, container destination, container size, container type, and the arrangement rule of container. The study case and playing rules in this game development are adjusted to contextual theory and real situation occured in container arrangement at ship bay. The prototype of this game is called StowaGame. StowaGame is a simulation-based educational games related to container arrangement in ship bay. This software are built with the integration of MBPP and entertainment concept. The effectivity of this educational game will be tested by several player. The MBPP concept of each player will be tested before and after playing the game. Paired-t test calculation shows that p-value is less than 0,05. Thus, it can be concluded that the educational game can educate people with container arrangement concept in ship bay.

Key words: educational games, master bay plan problem, container arrangement

PENDAHULUAN

Seiring dengan pesatnya permintaan akan transportasi laut, jasa pengiriman transportasi laut harus berpikir keras untuk meningkatkan efisiensi sehingga dapat meningkatkan daya saing terhadap jasa transportasi lain. Untuk mencapai hal tersebut, perlu adanya pengetahuan dan pemahaman terhadap perkembangan ilmu dan teknologi yang terkait. Akademisi dapat berperan aktif untuk memberikan kontribusi berupa transfer ilmu terhadap jasa transportasi laut. Salah satu metode transfer ilmu yang dapat dilakukan adalah menggunakan game edukasi.

Game edukasi berbasis simulasi didesain untuk mensimulasikan permasalahan yang ada sehingga diperoleh esensi atau ilmu yang dapat digunakan untuk menyelesaikan permasalahan tersebut. Game simulasi dengan tujuan edukasi ini dapat digunakan sebagai salah satu media edukasi yang memiliki pola pembelajaran *learning by doing*. Berdasarkan pola yang dimiliki oleh game tersebut, pemain dituntut untuk belajar sehingga dapat menyelesaikan permasalahan yang ada. Status game, instruksi, dan tools yang disediakan oleh game akan membimbing pemain secara aktif untuk menggali informasi sehingga dapat memperkaya pengetahuan dan strategi saat bermain.

Game edukasi sangat menarik untuk dikembangkan. Ada beberapa kelebihan dari *game* edukasi dibandingkan dengan metode edukasi konvensional. Salah satu kelebihan utama *game* edukasi adalah pada visualisasi dari permasalahan nyata. *Massachusetts Institute of Technology* (MIT) berhasil membuktikan bahwa *game* sangat berguna untuk meningkatkan logika dan pemahaman pemain terhadap suatu masalah melalui proyek *game* yang dinamai *Scratch* (<http://scratch.mit.edu>). Berdasarkan hasil penelitian-penelitian sebelumnya, telah dibuktikan *game* edukasi dapat menunjang proses pendidikan (Marsh, *et al.*, 2005; Clark, 2006).

Game edukasi unggul dalam beberapa aspek jika dibandingkan dengan metode pembelajaran konvensional. Salah satu keunggulan yang signifikan adalah adanya animasi yang dapat meningkatkan daya ingat sehingga siswa dapat menyimpan materi pelajaran dalam waktu yang lebih lama dibandingkan dengan metode pengajaran konvensional (Clark, 2006). Unsur lain yang dimasukkan dalam *game* yang dibuat adalah hiburan. Dengan adanya hiburan, diharapkan proses belajar dapat lebih menyenangkan sehingga siswa dapat belajar dengan lebih efektif. *Learning is most effective when it is fun* (Loomans and Kohlberg, 1993).

Pada penelitian ini, pengembangan *game* edukasi akan fokus pada permasalahan *loading* kontainer di suatu *bay* kapal. Permasalahan yang dihadapi dalam perencanaan *loading* di kapal biasa disebut dengan *Master Bay Plan Problem* (MBPP) atau *Stowage Planning Problem* (SPP). Kompleksitas dari permasalahan MBPP dapat dilihat dari konstrain yang ada, yaitu berat kontainer, tujuan kontainer, ukuran kontainer, tipe kontainer, dan aturan peletakan kontainer.

Pada praktek di lapangan, penempatan kontainer pada suatu *bay* kapal ditentukan oleh pangkalan kapal yang berkolaborasi dengan pekerja yang bertanggung jawab atas peletakan kontainer pada kapal yang dimaksud. Dengan berpedoman pada rancang bangun *bay* kapal, pekerja tersebut dapat menentukan peletakan kontainer yang sesuai dengan tipenya yang beragam. Perencanaan ini bertujuan untuk mengedukasi pemain sehingga dapat meminimasi terjadinya *shifting* pada kontainer yang tidak dibongkar di pelabuhan berikutnya. Agar keputusan yang diambil tepat, pengalaman dan pengetahuan yang tinggi adalah hal yang mutlak harus dimiliki oleh pekerja. Metode optimasi untuk meminimasi jumlah *shifting* kontainer telah dibahas oleh beberapa peneliti sebelumnya, misalnya Ambrosino *et al.*, (2004).

Penelitian ini secara spesifik akan menfokuskan pada penterjemahan pengetahuan-pengetahuan mengenai MBPP atau SPP ke dalam suatu *game* edukasi yang dapat menjadi media latihan dan pembelajaran mengenai permasalahan penataan kontainer di *bay* kapal. *Game* ini kemudian dinamakan *StowaGame*.

PERANCANGAN GAME

Konsep Dasar

StowaGame dibuat untuk digunakan untuk mahasiswa yang mendalami bidang manajemen logistik dan manajemen rantai pasok (*supply chain management*). *StowaGame* dirancang untuk dapat mensimulasikan suatu kondisi permasalahan yang terjadi dalam penataan kontainer di *bay* kapal. Pemain diharuskan untuk menyelesaikan dan memberikan solusi terhadap permasalahan tersebut. *StowaGame* dirancang sedemikian rupa sehingga dapat membangun pemikiran heuristik dari pemain sehingga dapat mengembangkan imajinasi dan membangun pemikiran secara sistemik terhadap permasalahan SPP sehingga dapat dicari solusi yang diterapkan. Titik berat dari perancangan *StowaGame* adalah untuk memberikan pengalaman dalam menghadapi suatu permasalahan dan mencoba mencari solusi terhadap permasalahan tersebut.

Dengan adanya simulasi dalam *game* yang dibuat, pemain dihadapkan pada permasalahan penataan kontainer yang relatif mirip dengan keadaan sebenarnya. Misalnya, pemain akan mengetahui bagaimana penataan kontainer atau *box* untuk kapal atau kendaraan multi tujuan. Dalam hal ini, pengajar akan berperan sebagai pendamping dan pemain akan berperan aktif dalam usahanya menyelesaikan permasalahan yang ada.

Unsur pedagogis adalah salah satu dari unsur yang terdapat pada konsep dari *StowaGame*. Pedagogis yang dimaksud disini adalah terjadinya *student-centered learning* sehingga siswa yang akan mencari tahu apa permasalahan yang terjadi dan bagaimana solusinya. Dalam menjalankan permainan, rancangan penataan kontainer yang diberikan oleh pemain dievaluasi oleh *game* secara terus menerus. *Game* akan memberikan informasi mengenai status dari kontainer yang telah ditata pada *bay* kapal.

Evaluasi yang dilakukan oleh *game* adalah waktu yang dibutuhkan dalam menyelesaikan permasalahan, benar atau tidaknya penataan, dan jumlah *unnecessary shifting* yang terjadi. Evaluasi yang diberikan oleh *game* akan memberikan umpan

balik kepada pemain sehingga pemain dapat mengerti kesalahan yang dilakukan sehingga pemain dapat memperbaiki kesalahannya.

Selain itu, hal lain yang terkandung dalam *StowaGame* ini adalah adanya unsur hiburan dalam rancangan *user interface*. Dengan adanya unsur hiburan yang terkandung di dalam *game* ini, pemain diharapkan lebih mampu menyerap informasi yang disampaikan. Salah satu keuntungan lain dari adanya unsur hiburan adalah pemain tidak akan cepat bosan dalam menerima informasi yang akan diberikan.

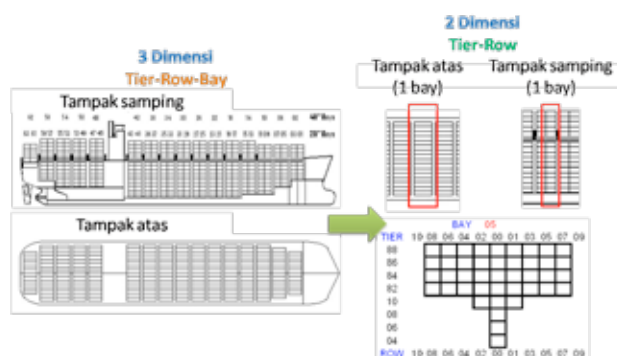
Simplifikasi

StowaGame ini dirancang untuk menyerupai permasalahan nyata dari penataan kontainer di *bay* kapal menjadi suatu permasalahan yang relatif lebih simpel namun tidak meninggalkan aspek-aspek sebenarnya dari penataan kontainer. Tujuan dari simplifikasi dari permasalahan nyata tersebut adalah untuk lebih mempermudah penjelasan dan transfer pengetahuan dari pembuat *game* terhadap pemain *game*. Simplifikasi dari permasalahan tersebut juga dapat digunakan untuk memudahkan pemrograman dari *game*. Simplifikasi dari permasalahan penataan kontainer dalam *game* ini antara lain adalah dimensi *bin-packing*, pemindahan kontainer yang telah diletakkan, dan parameter yang menyebabkan terjadinya kemiringan dari kapal.

Permasalahan penataan kontainer di *bay* kapal adalah permasalahan *bin-packing* tiga dimensi yang terdiri dari dimensi *bay*, *tier*, dan *row*. Dalam *game* yang dibuat, permasalahan penataan kontainer tersebut disimplifikasikan menjadi permasalahan *bin-packing* dua dimensi yang hanya terdiri dari dimensi *tier* dan *row*. Dalam hal ini dimensi *bay* masih terlibat di dalam permasalahan *game* yaitu sebanyak satu *bay* namun ukuran panjang kontainer adalah sama dengan panjang *bay* yang mengakibatkan dimensi *bay* menjadi tidak perlu diperhitungkan. Simplifikasi dari dimensi permasalahan *bin-packing* dapat dilihat pada gambar 1.

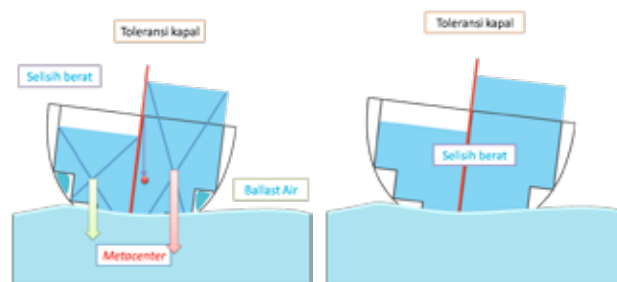
Simplifikasi lain yang terdapat dalam *game* ini yaitu kontainer yang telah diletakkan di suatu slot pada *bay* kapal tidak boleh dipindahkan ke slot lain. Hal ini dimaksudkan untuk meningkatkan tantangan yang ada. Jika pemain melakukan kesalahan peletakan kontainer, pemain dapat memanfaatkan fasilitas *undo last move* yang disediakan dalam *game*. Fasilitas ini dapat digunakan berkali-kali tetapi hanya berlaku untuk *undo* satu langkah terakhir saja.

Simplifikasi yang terakhir yaitu parameter yang menyebabkan terjadinya kemiringan kapal.



Gambar 1. Simplifikasi Dimensi *Bin-Packing*

Kemiringan kapal dapat terjadi akibat selisih berat, *ballast air*, *meta center*, dan toleransi dari masing-masing kapal. Batas maksimum kemiringan kapal umumnya adalah ± 15 derajat dari titik normal. Kemiringan kapal adalah positif jika searah dengan jarum jam dan negatif jika berlawanan dengan arah jarum jam. Dalam *game* ini, parameter yang menyebabkan terjadinya kemiringan kapal adalah toleransi dari kapal dan selisih berat saja. Simplifikasi dari parameter kemiringan kapal dapat dilihat pada gambar 2.



Gambar 2. Simplifikasi dari Parameter Penyebab Kemiringan Kapal

Pemain dihadapkan pada penataan sejumlah kontainer yang telah ditentukan sedemikian rupa sehingga tidak ada *unnecessary shifting* dari kontainer di *yard* dalam penataannya ke dalam *bay* kapal. Pemain harus mengambil kontainer yang terletak di bagian paling atas dari tumpukan kontainer yang ada. Dalam hal ini, selain variabel-variabel yang melingkupi penataan itu sendiri, ada variabel lain yang berupa limit waktu. Dengan adanya limit waktu tersebut maka penataan kontainer harus dilaksanakan tidak melebihi waktu yang telah ditentukan.

Pemain juga akan dihadapkan pada permasalahan kapal *direct* dan kapal *feeder*. Pemain akan menata kontainer untuk satu, dua atau tiga tujuan sesuai dengan tahapan kesulitan yang telah ditentukan.

Untuk penataan kontainer lebih dari satu tujuan, pemain harus memenuhi batasan *unnecessary shifting* maksimum yang diperbolehkan.

Selama permainan berlangsung, pemain akan diberikan status penataan dan list kontainer yang masih tersedia. Status penataan yang dimaksud adalah kemiringan kapal, berat di kiri dan kanan kapal, tinjauan hasil penataan (untuk semua variabel) dan waktu yang tersedia.

Rancangan Cerita

Game edukasi berbasis simulasi harus memiliki cerita *game* yang sangat kuat di mana pemain yang awam (tidak memiliki dasar pengetahuan terhadap permasalahan) dapat mengerti permasalahan yang ada dengan adanya tuntunan dari cerita *game*. Salah satu kekuatan utama dari edukasi adalah cerita *game*.

Cerita *game* tersebut harus dibuat semenarik dan sejelas mungkin agar pemain dapat mengerti apa yang harus dilakukan olehnya untuk menyelesaikan permasalahan yang ada dengan bantuan dan supervisi yang minimal dari orang lain atau tutor. Cerita *game* disini harus menarik agar pemain tidak bosan membaca tiap kalimat yang ada dalam cerita *game*. Penjelasan terhadap masalah yang ada juga harus dimasukkan dalam cerita *game* tersebut.

Beberapa hal yang dimasukkan dalam cerita *game* tersebut adalah:

1. Langkah-langkah permainan
2. Kontrol permainan
3. Cerita singkat permasalahan (*overview*)
4. Cerita detail permasalahan

Cerita dari *game* tersebut akan dimunculkan dengan menyesuaikan terhadap kebutuhan dari pemain. Cerita singkat permasalahan, langkah-langkah penyelesaian serta kontrol permainan akan dijelaskan pada awal permainan sedangkan cerita detail permasalahan akan dimunculkan pada tiap awal *level*. Hal ini dimaksudkan agar pemain dapat mencerna informasi dan pengetahuan yang diberikan dengan mudah. Informasi mengenai penataan dari kontainer tidak dijelaskan sekaligus tetapi menyesuaikan dengan desain dari *level* yang telah ditentukan. Hal ini dimaksudkan agar pemain dapat mengingat dan memahami dengan mudah informasi yang diperoleh.

Rancangan Evaluasi Terhadap Pemain

Pemain akan diberikan *feedback* jika gagal dalam menyelesaikan suatu permasalahan. *Feedback* ini akan diberikan secara otomatis dan disesuaikan dengan kesalahan yang dilakukan oleh pemain.

Pemain dapat memahami kesalahannya dan menggunakan saran yang dihasilkan oleh sistem dalam usahanya untuk menyelesaikan permasalahan yang ada. Saran yang dibuat bersifat *general* terhadap kesalahan atau permasalahan yang dihadapi.

Rancangan Level Permainan

Desain *level* yang baik mutlak harus dimiliki oleh sebuah *game* edukasi. Dengan desain *level* yang baik, pemain akan lebih mudah dalam memahami permasalahan yang ada. *Level* itu sendiri dapat diartikan sebagai tahapan edukasi, mulai dari yang paling mudah sampai ke tingkat yang tersulit dalam *game*. Jika desain *level* tersebut tidak diatur dengan baik, alurnya melompat-lompat atau tidak menyesuaikan dengan tingkat pemahaman dari pemain, pemain akan kesulitan menangkap informasi yang diberikan oleh *game*. Desain *level* dari *game* yang akan dibuat dengan mempertimbangkan variabel penataan dan tingkat kesulitan dari penataan kontainer di *bay* kapal dapat dilihat pada tabel 1.

Tabel 1. Desain Level Permainan

Level	Konstrain				
	Kontainer			Kapal	
	Tujuan	Tipe	Ukuran	Berat	Keseimbangan
Pemula					
Level 1	Satu tujuan	3	20"	x	x
Level 2	Satu tujuan	3	20"	Ñ	x
Level 3	Satu tujuan	3	20"	Ñ	Ñ
Level 4	Satu tujuan	3	20" dan 40"	Ñ	Ñ
Menengah					
Level 5	Dua tujuan	3	20"	Ñ	Ñ
Level 6	Dua tujuan	3	20"	Ñ	Ñ
Level 7	Tiga tujuan	3	20" dan 40"	Ñ	Ñ

Desain *level* tersebut juga dibagi menjadi dua tingkat kesulitan utama yaitu pemula dan menengah. Tingkat kesulitan tersebut dibagi berdasarkan pada tujuan dari kontainer. Tingkat kesulitan pemula adalah penataan kontainer untuk satu tujuan sedangkan tingkat kesulitan menengah adalah penataan kontainer untuk lebih dari satu tujuan. Adanya perbedaan tingkat kesulitan tersebut adalah dikarenakan ada parameter penilaian tambahan pada penataan kontainer lebih dari satu tujuan yaitu *unnecessary shifting*.

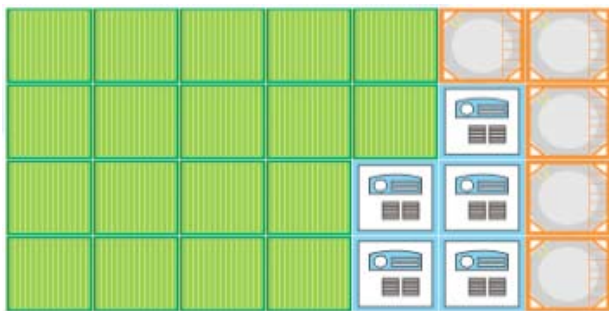
PENGEMBANGAN PERANGKAT LUNAK

Perangkat lunak *game* edukasi ini dibuat pada *platform*.Net Framework + XNA dengan bahasa pemrograman C# atau dengan kata lain adalah

sebuah *game* yang dapat dijalankan pada komputer yang berbasis Windows.

Rancangan Input *Game*

Kontainer yang berada di lapangan kontainer (*container yard*) di pelabuhan yang akan dipindahkan ke *bay* kapal dirancang tetap. Pemilihan sifat *list* kontainer yang tetap ini (tidak acak) dikarenakan agar permasalahan dari penataan kontainer tersebut dapat diperbandingkan antara pemain satu dengan pemain lainnya. Selain itu, penataan kontainer yang bersifat acak akan jauh lebih sulit dalam pemrogramannya karena penataan kontainer di *yard* itu juga memiliki aturan penataan kontainer. Set dari kontainer yang terdapat di *yard* sudah diatur sebelumnya. Terdapat beberapa set permasalahan penataan dibuat. Masing-masing set permasalahan kontainer tersebut sudah dilengkapi dengan lima variabel yang terkait dengan penataan kontainer. Contoh dari set penataan di lapangan kontainer yang merupakan input dari permainan dapat dilihat di gambar 3.



Gambar 3. Penataan di Lapangan Kontainer sebagai Input Permainan

Rancangan Komponen Pendukung *Game*

Komponen pendukung dalam *game* adalah desain gambar, suara, dan cerita *game*, yang dapat membantu penyampaian informasi dari *game* terhadap pemain. Bahan pendukung yang dimaksud memiliki fungsi utama yaitu untuk tambahan *knowledge* atau *entertainment* saja. Bahan pendukung ini utamanya adalah kosmetik dalam *game* namun hal ini adalah salah satu hal penting yang tidak boleh diremehkan karena adanya desain yang menarik dapat meningkatkan minat dari pemain *game* dan dapat memperlambat munculnya kebosanan. Desain pendukung utama dari *game* yang dibuat adalah desain *bay* dan bentuk kapal, desain *background*, dan desain tipe kontainer. Bahan pendukung dari *game* tidak hanya tampilan dari *game* tetapi juga suara dan animasi dari *game*.

Rancangan Menu *Game*

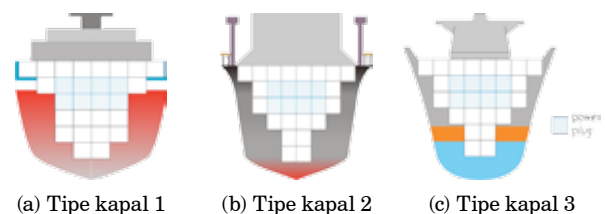
Menu *game* adalah tampilan utama dari *game* yang dimainkan. Menu *game* yang ditampilkan berisi tentang *new game*, *controls*, *introduction*, *game story*, dan *exit game*. Desain tampilan menu *game* dapat dilihat pada gambar 4.



Gambar 4. Desain Menu *Game*

Rancangan *Bay* dan Bentuk Kapal

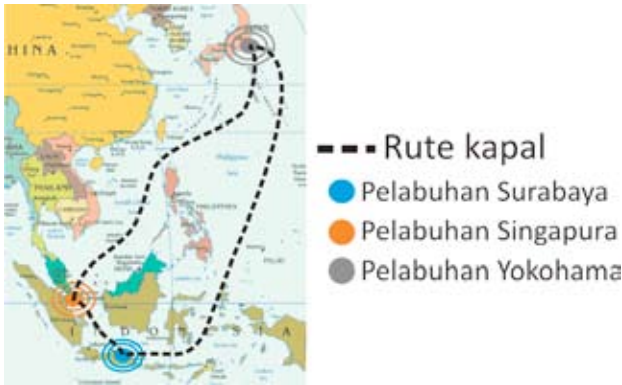
Ada tiga kapal yang dapat dimainkan oleh pemain. Masing-masing kapal memiliki perbedaan desain *bay* dan bentuk kapal. Dengan adanya perbedaan *bay* dan bentuk kapal yang telah dijelaskan sebelumnya, maka perlu adanya desain yang berbeda dari masing-masing kapal untuk memperjelas perbedaan dari masing-masing kapal. Adanya perbedaan bentuk *bay* dalam desain kapal tidak memengaruhi tingkat kesulitan dari penataan secara signifikan. Adanya beberapa bentuk *bay* dan kapal hanya dimaksudkan untuk memperkaya konten dari *game* yang dikembangkan. Desain dari *bay* dan bentuk kapal dapat dilihat pada gambar 5.



Gambar 5. Rancangan *Bay* dan Bentuk Kapal

Rancangan *Background* Dari Pelabuhan

Kapal memiliki tiga buah tujuan pelabuhan yaitu Singapura, Yokohama, dan Surabaya. Tujuan dari pelabuhan kapal tersebut akan diinformasikan terhadap pemain dalam cerita *game* yang dibuat. Rute dari kapal dapat dilihat pada gambar 6.



Gambar 6. Tujuan Kapal

Pemain akan mulai memainkan *game* pada pelabuhan di kota Surabaya. Urutan dari pelabuhan tujuan *default*-nya adalah Surabaya - Singapura - Yokohama. Urutan ini pada suatu saat akan berganti sesuai dengan *level* dari permainan yang telah dicapai oleh pemain. *Background* dari pelabuhan tujuan dari kapal dibuat sesuai dengan persepsi dari peneliti terhadap pelabuhan tujuan yang disesuaikan dengan kondisi pelabuhan sebenarnya. Desain *background* dibuat semenarik mungkin tujuannya agar dapat membuat pemain tertarik untuk memainkan *game* ini. *Background* ini digunakan sebagai gambar yang merepresentasikan keadaan pelabuhan di tempat tujuan. Desain *background* pelabuhan dapat dilihat pada gambar 7.



Gambar 7. Desain *Background* Pelabuhan

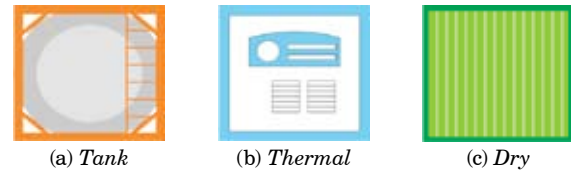
Rancangan Kontainer

Variabel yang terkait dengan kontainer ada empat macam yaitu tipe, ukuran, berat, dan tujuan. Masing-masing informasi tersebut akan digambarkan dalam *game* sesuai dengan variabel yang terlibat pada masing-masing kontainer. Pemain juga akan diberikan informasi tersebut sesuai dengan kontainer yang dipilih.

Desain Informasi Tipe Kontainer

Tipe kontainer yang terlibat dalam permasalahan penataan adalah kontainer dengan tipe *dry*,

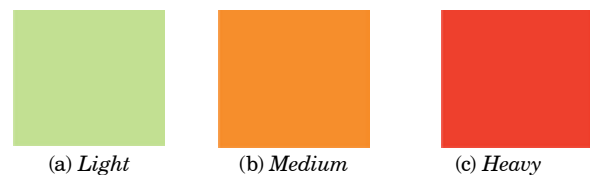
thermal, dan *tank*. Tipe kontainer tersebut harus direpresentasikan dengan gambar yang sesuai. Desain kontainer yang dibuat disesuaikan dengan tipe kontainer dan bentuk kontainer sebenarnya. Desain tipe kontainer dapat dilihat pada gambar 8.



Gambar 8. Desain Tipe Kontainer

Desain Informasi Kelas Bobot Kontainer

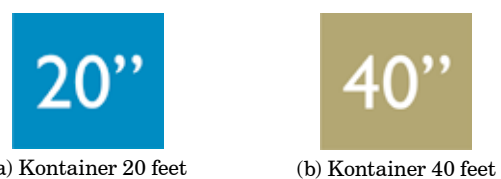
Kelas bobot dari kontainer yang terlibat dalam permasalahan penataan adalah kelas *light*, *medium*, dan *heavy*. Informasi dari kelas berat sebuah kontainer direpresentasikan dengan gambar yang sesuai. Desain informasi berkenaan dengan kelas berat kontainer dapat dilihat pada gambar 9.



Gambar 9. Desain Kelas Bobot Kontainer

Desain Informasi Ukuran Kontainer

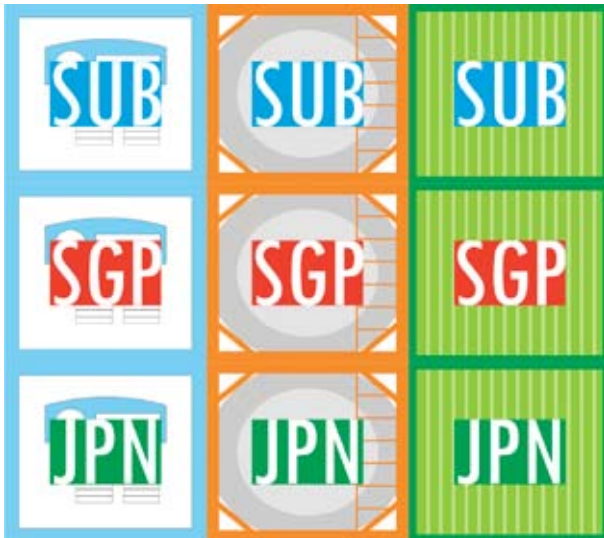
Ukuran dari kontainer yang terlibat dalam permasalahan penataan di penelitian ini adalah kontainer dengan ukuran 20 feet dan 40 feet. Informasi dari ukuran sebuah kontainer direpresentasikan dengan gambar yang sesuai. Desain informasi berkenaan dengan ukuran kontainer dapat dilihat pada gambar 10.



Gambar 10. Desain Informasi Ukuran Kontainer

Desain Informasi Tujuan Kontainer

Tujuan dari kontainer yang terlibat dalam permasalahan penataan di penelitian ini adalah kontainer dengan tujuan Jepang, Surabaya, dan Singapura. Informasi dari tujuan kontainer direpresentasikan dengan gambar yang sesuai. Desain informasi berkenaan dengan tujuan kontainer dapat dilihat pada gambar 11.



Gambar 11. Desain Informasi Tujuan Kontainer

Desain Informasi Kontainer Terpilih

Sistem *game* akan memberikan informasi dari kontainer yang dipilih atau ditunjuk oleh kursor *mouse*. Dibandingkan dengan desain informasi sebelumnya yang menjelaskan masing-masing *state* dari variabel yang ada, informasi ini akan menjelaskan secara rinci variabel *state* yang terlibat dari kontainer yang ditunjuk oleh kursor. Desain informasi kontainer terpilih dapat dilihat pada gambar 12.

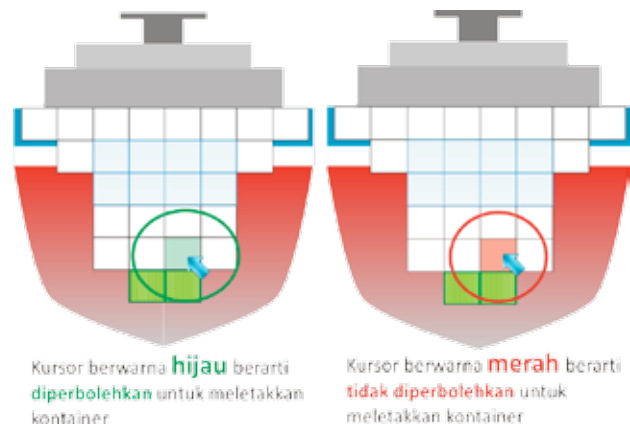


Gambar 12. Desain Informasi Variabel Kontainer Keseluruhan

Dalam contoh gambar di atas, kontainer yang ditunjukkan adalah kontainer *dry*. Pada *game* yang dikembangkan, semua tipe kontainer yaitu *dry*, *thermal* dan *tank* akan ditunjukkan dan digambarkan. Informasi dari tipe, berat, ukuran dan tujuan dari kontainer disesuaikan dengan aturan variabel dalam *level* dan kontainer yang dipilih oleh pemain.

Rancangan Informasi Peletakan Kontainer

Kontainer yang terpilih dan akan diletakkan di kapal akan diberikan dapat menghasilkan dua keputusan yaitu boleh ditaruh atau tidak boleh ditaruh. Sistem dari *game* akan menghasilkan keputusan terkait dengan peletakan kontainer oleh pemain. Pemain akan diberikan informasi terkait keputusan tersebut dengan gambar yang sesuai (gambar 13). Kursor dari kontainer akan berwarna hijau jika pemain meletakkan kontainer sesuai dengan batasan yang ada dan kursor akan berwarna merah jika pemain meletakkan kontainer tidak mengikuti batasan yang ada.



Gambar 13. Desain Informasi Variabel Kontainer Keseluruhan

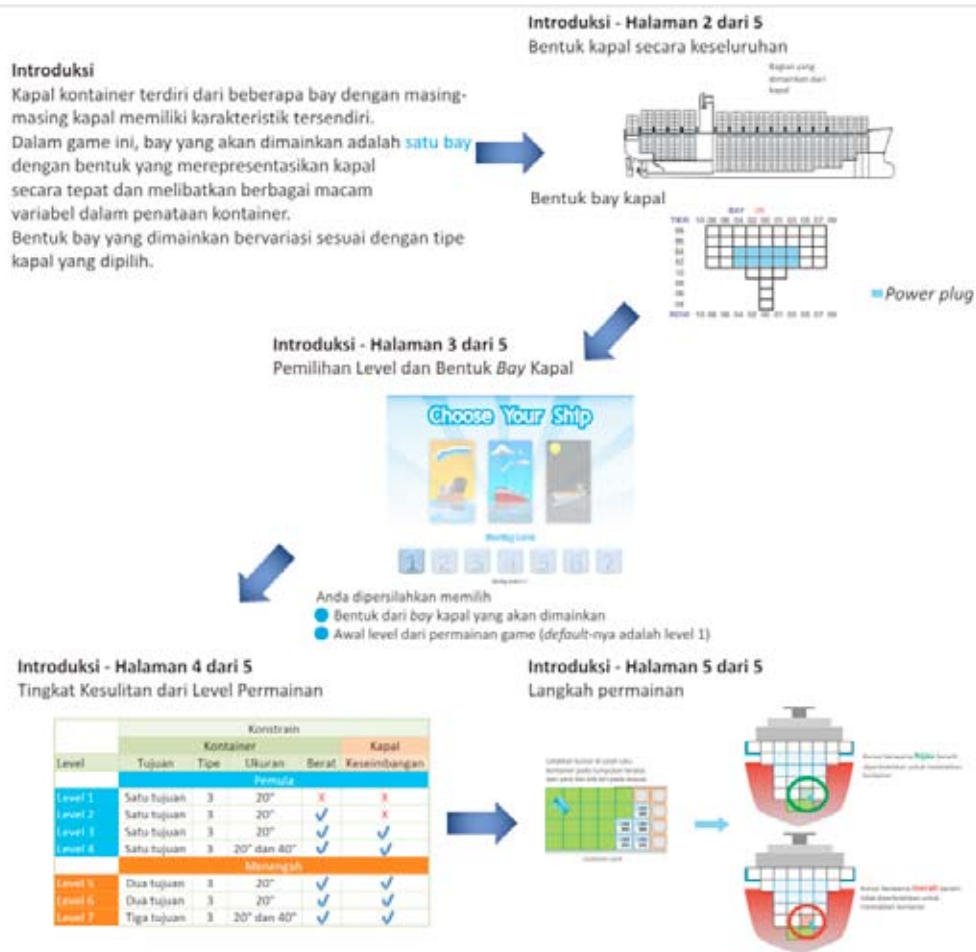
Rancangan Cerita *Game*

Cerita *game* dibuat dalam bahasa Indonesia. Cerita *game* ditampilkan secara menyeluruh pada menu utama dan ditampilkan secara bertahap untuk tiap-levelnya. Cerita *game* disesuaikan dengan batasan dan informasi yang terdapat di dalam permasalahan yang diangkat pada tiap levelnya. Introduksi dari *game* juga dijelaskan dalam cerita *game*. Introduksi dari *game* umumnya berisi *overview* dan langkah-langkah permainan dalam *game*.

Rancangan Aturan Permainan

Menu Pemilihan Kapal

Bentuk kapal dan *level* permulaan dapat dipilih sesuai dengan keinginan dari pemain. Bentuk kapal yang tersedia ada tiga bentuk kapal dengan tujuh *level* yang dapat dipilih. Implementasi dari menu pemilihan kapal dapat dilihat pada gambar 15.



Gambar 14. Cerita Introduksi Game



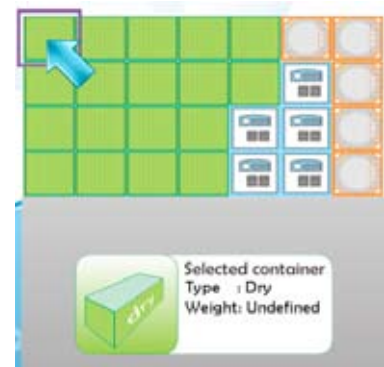
Gambar 15. Menu Pemilihan Kapal

Pemain dapat memilih untuk memulai *game* pada salah satu dari tujuh level yang tersedia. *Default* dari level permulaan dari *game* adalah level satu.

Aturan Pemilihan Kontainer

Kontainer yang terdapat di *yard* dapat dipilih oleh pemain untuk dimasukkan ke *bay* kapal.

Batasan dari kontainer yang dapat dipilih adalah kontainer tersebut ada di atas tumpukan kontainer yang berada di *yard*. Pemain tidak diperbolehkan untuk memilih kontainer yang terletak di bagian tengah atau bagian bawah dari tumpukan. Implementasi dari aturan pemilihan kontainer dapat dilihat pada gambar 16.



Gambar 16. Pemilihan Kontainer

Kontainer yang dipilih oleh pemain akan diberi sebuah kotak untuk menandai bahwa pemain telah memilih sebuah kontainer. Pemain dapat melakukan *cancel* terhadap pemilihan dari kontainer dengan tombol yang telah dijelaskan pada menu kontrol. Jika pemain tidak memilih kontainer yang berada di tumpukan kontainer maka hal tersebut tidak akan diijinkan oleh sistem.

Aturan Penataan Kontainer Berdasarkan Tipe

Kontainer yang telah dipilih oleh pemain harus ditata sesuai dengan aturan tipe dari kontainer. Kontainer *thermal* harus diletakkan di lokasi *bay* yang memiliki *power plug* (gambar 17), kontainer *tank* tidak boleh ditumpuk dengan kontainer lain (gambar 18) dan kontainer *dry* boleh diletakkan di mana saja di lokasi *bay* kapal (gambar 19).

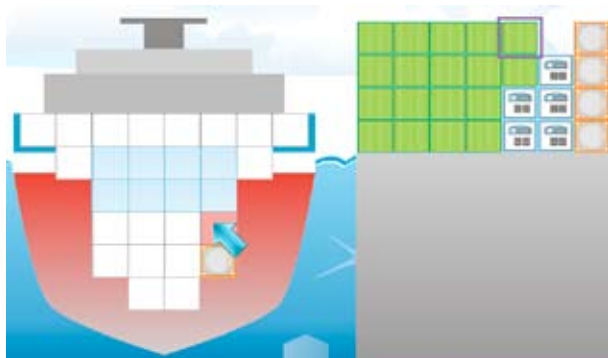


(a) Kontainer *thermal* diletakkan di bay kapal yang tidak memiliki *power plug*

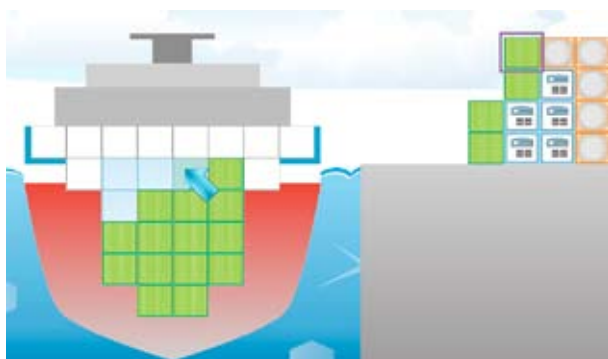


(b) Kontainer *thermal* diletakkan di bay kapal yang memiliki *power plug*

Gambar 17. Penataan Kontainer *Thermal*



Gambar 18. Penataan Kontainer *Tank*



Gambar 19. Penataan Kontainer *Dry*

Implementasi aturan penataan kontainer adalah dasar awal dari penataan kontainer di *bay* kapal yang berhubungan dengan *level* atau tingkat kesulitan penataan selanjutnya.

Aturan Penataan Kontainer Berdasarkan Bobot

Penempatan kontainer harus memperhatikan kontainer di bawahnya. Kontainer yang ditumpuk harus memiliki kelas bobot yang sama atau lebih tinggi dari kontainer di atasnya. Jika hal tersebut tidak dipatuhi maka sistem *game* akan melarang penempatan kontainer tersebut (gambar 20).



(a) Peletakan yang salah



(b) Peletakan yang benar

Gambar 20. Aturan Penataan Berdasarkan Bobot

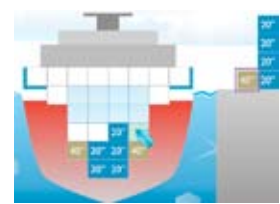
Gambar 20a tersebut menunjukkan bahwa kontainer dengan kelas bobot *heavy* tidak diperbolehkan untuk diletakkan di atas kontainer dengan kelas bobot *medium* sedangkan gambar 4.20b menunjukkan sebuah peletakan yang benar di mana kontainer dengan kelas bobot *heavy* diletakkan di atas kontainer dengan kelas berat yang sama.

Aturan Penataan Kontainer Berdasarkan Ukuran

Ukuran 20 *feet* dan 40 *feet* yang terdapat dalam permasalahan penataan diletakkan sesuai dengan aturan yang berlaku. Kontainer dengan ukuran 40 *feet* tidak diperbolehkan untuk diletakkan di atas kontainer dengan ukuran 20 *feet* tetapi kontainer dengan ukuran 20 *feet* dapat diletakkan di atas kontainer dengan ukuran 40 *feet*. Kontainer dengan ukuran 40 *feet* hanya diperbolehkan untuk diletakkan di atas kontainer dengan yang sama (lihat gambar 21).



(a) Penataan kontainer 40 feet di atas kontainer 20 feet



(b) Penataan kontainer 40 feet di atas kontainer 40 feet

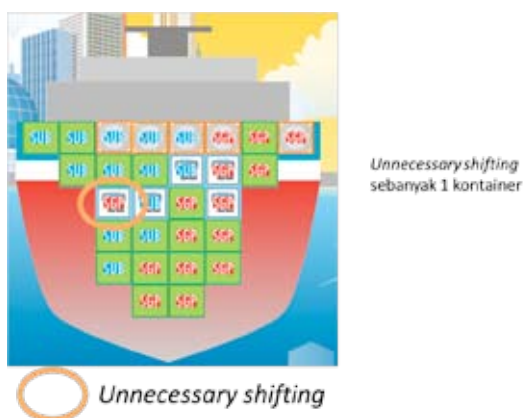
Gambar 21. Aturan Penataan Berdasarkan Ukuran Kontainer

Aturan Penataan Kontainer Berdasarkan Kemiringan Kapal

Penataan kontainer pada suatu *state* pastinya akan menghasilkan suatu kemiringan terhadap kapal. Jika kemiringan tersebut terjadi karena selisih berat dari suatu sisi yang melebihi toleransi kapal maka pemain harus meletakkan kontainer pada sisi yang lainnya. Pemain tidak diperbolehkan untuk menambah kontainer pada sisi yang mengakibatkan kemiringan dari kapal.

Perhitungan *Unnecessary Shifting*

Unnecessary shifting dari kontainer hanya terjadi untuk permasalahan penataan di atas level empat. *Unnecessary shifting* yang dihitung adalah *shifting* per kontainer yang berada di atas kontainer pada pelabuhan bongkar di mana kontainer yang menumpuk tersebut adalah kontainer pada pelabuhan berikutnya. Implementasi dari perhitungan *unnecessary shifting* dapat dilihat pada gambar 22.



Gambar 22. Perhitungan *Unnecessary Shifting*

Penilaian *unnecessary shifting* dari gambar di atas adalah sebanyak 1 (satu) kontainer. Kontainer yang mengalami *unnecessary shifting* dari gambar di atas adalah kontainer Singapura yang ditumpuk di atas kontainer dengan tujuan Surabaya.

Aturan Evaluasi Status "Sukses" atau "Gagal"

Penataan yang dilakukan oleh pemain akan diberikan status penataan sesuai dengan aturan yang telah ditentukan. Penataan akan diberi status gagal jika pemain tidak dapat menata semua kontainer di *yard*. Sistem *game* akan menghasilkan status penataan yang sukses jika pemain dapat menata semua kontainer ke dalam kapal.

UJI EDUKASI DARI GAME

Tahap uji edukasi dari *game* mencakup tentang perancangan soal dan uji edukasi dari *game*. Soal

atau pertanyaan terkait dengan edukasi dari *game* disesuaikan dengan informasi yang diberikan oleh *game*. Uji edukasi dari *game* dilakukan dengan *pre-test* dan *post-test* untuk pertanyaan yang sama. Edukasi dari *game* akan dibuktikan secara statistik dengan uji *paired-t test*.

Perancangan soal

Soal dari *game* edukasi ini adalah soal pilihan ganda yang terkait dengan informasi yang diberikan pada *game*. Soal yang dibuat terdiri dari 13 soal dengan urutan soal yang disesuaikan tingkatan atau *level* yang terdapat dalam *game*. Ada beberapa alternatif dalam pembuatan soal, tidak hanya pilihan ganda, yaitu esai, dan pernyataan benar atau salah. Alternatif pilihan ganda dipilih karena dapat mempermudah penilaian edukasi dan juga dengan adanya pilihan ganda tingkat kesulitan dari soal relatif sedang karena adanya pilihan jawaban (salah) yang relatif mendekati jawaban yang benar.

Uji Edukasi

Uji edukasi perlu dilakukan agar sebuah *game* dapat dikatakan bahwa *game* tersebut dapat meningkatkan pengetahuan dari pemain terhadap suatu permasalahan. Uji edukasi yang dilakukan adalah *pre-test* dan *post-test*. *Pre-test* adalah uji pengetahuan dari pemain sebelum memainkan *game* sedangkan *post-test* adalah uji pengetahuan dari pemain setelah memainkan *game*. Soal dari *pre-test* dan *post-test* yang diberikan adalah sama. Hal yang akan diperbandingkan adalah jumlah nilai benar dari *test* yang diberikan. Hasil dari uji edukasi dapat dikatakan bahwa *game* dapat meningkatkan pengetahuan pemain jika perbedaan jumlah nilai benar dari *test* yang diberikan berbeda secara signifikan berdasarkan pada uji statistik.

Dalam uji edukasi ini akan dilakukan uji *paired-t* dengan faktor permainan *game*. Pemain akan mengerjakan soal yang telah disebutkan di atas sebelum dan sesudah memainkan *game*. Yang akan dibandingkan dalam uji anova ini adalah jumlah jawaban benar dari soal dengan faktor *treatment* permainan *game*. *Game* akan dinyatakan memiliki faktor edukasi jika jumlah jawaban benar dari pemain berbeda secara signifikan berdasar uji statistik dengan menggunakan *paired-t test*.

Responden dari *game* ini ada 15 orang yang diambil dengan metode judgment sampling di mana peneliti mengetahui tingkat pemahaman dari responden terkait dengan penataan kontainer di bay kapal. Responden mengerjakan soal yang diberikan sesuai dengan ketentuan dari peneliti. Hasil dari uji edukasi terhadap responden dapat dilihat pada tabel 2.

Tabel 2. Hasil Uji Edukasi terhadap Responden

No	Responden	Jumlah soal yang berhasil di jawab	
		Pre-Test	Post-Test
1	Arum Puji Lestari	3	12
2	Titi Anindhita	0	13
3	Putri	0	13
4	Worosita Mulyaninglisti	1	13
5	Habib Hamidi	1	13
6	Kusbiantoro	2	13
7	Nadia Aulia Arifin	0	13
8	Savira Evany	3	12
9	Nadya Rahmaviany	1	13
10	Widha Kusuma	2	12
11	Ratih Ardia Sari	3	13
12	Indah Baroroh	2	13
13	Hutama Purwananta	2	12
14	Agung Puguh	2	11
15	Andra Wibisono	3	13

Hasil dari uji edukasi terhadap responden tersebut diolah dengan menggunakan uji *paired-t test* untuk mengetahui apakah *game* yang dibuat dapat memberikan edukasi terhadap pemainnya.

Hipotesis dari uji *paired-t test* yang akan dilakukan adalah:

H_0 = *Game* yang dibuat tidak memberikan edukasi terhadap pemain

H_1 = *Game* yang dibuat memberikan edukasi terhadap pemain

Hipotesis awal diterima jika $p - value \geq 0.05$ dan hipotesis tandingan diterima jika $p - value < 0.04$. Perhitungan dari *paired-t test* terkait dengan uji edukasi di atas (tabel 3) yaitu:

Tabel 3. Rataan Sampel

	Pre-test	Post-test
Rata-rata	1,666667	12,6
Standar deviasi	1,112697	0,6324555
Jumlah sampel	15	15
Ho	0	
df	14	
t Stat	-28,4871	
p-value (two-tail)	8,52E-14	
t Critical two-tail	2,144787	

Dari perhitungan *paired-t test* yang telah dilakukan (tabel 3) diperoleh $p - value < 0.05$ sehingga kesimpulan yang diambil adalah tolak H_0 dan nyatakan bahwa *game* yang dibuat memberikan perbedaan yang signifikan terhadap pengetahuan pemain terkait dengan permasalahan penataan kontainer di *bay* kapal.

SIMPULAN

Berdasarkan perancangan dan pengujian serta analisa dalam penelitian ini, maka kesimpulan yang diperoleh yaitu:

1. Dengan perancangan dan implementasi yang telah dilakukan, penelitian ini berhasil membuat suatu perangkat lunak berupa *game* edukasi untuk penataan kontainer di *bay* kapal yang diberi nama *StowaGame*. *StowaGame* telah mampu menggambarkan permasalahan penataan kontainer di *bay* kapal dengan simplifikasi yang telah ditentukan.
2. Uji *paired-t test* terhadap *game* menghasilkan $p - value < 0.05$. Hal ini dapat diartikan bahwa perangkat lunak yang dibuat mampu memberikan edukasi terhadap pemain *game*.

DAFTAR PUSTAKA

- Ambrosino, D., Sciomachen, A., and Tanfani, E., 2004. Stowing a Containership: The Master Bay Plan Problem. *Transportation Research A* 38, 81-99.
- Clark, D., 2006. *Game* and e-learning. Sunderland: Caspian Learning. URL:<http://caspianlearning.co.uk>. Akses terakhir: 01 April 2009.
- Loomans, D. and Kohlberg, K., 1993. *The Laughing Classroom*. Kramer, H.J. Inc.
- Marsh, T., Wong, W.L., Carriazo, E., Nocera, L., Yang, K., Varma, A., Yoon, H., Huang, Y., Kyriakakis, C., and Shahabi, C., 2005. User Experiences and Lessons Learned from Developing and Implementing an Immersive *Game* for the Science Classroom. *Proceedings of HCI International 2005*, Las Vegas, Nevada, USA.