

RANCANG BANGUN GAME EDUKASI UNTUK PENGENALAN DASAR LOGIKA ALGORITMA BERBASIS MOBILE

Paulus Lucky Tirma Irawan¹⁾, Felix Tandiono²⁾, Hendry Setiawan³⁾

^{1), 2), 3)} Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Ma Chung
 Jl Villa Puncak Tidar N-01, Tidar, Malang
 Email : paulus.lucky@machung.ac.id¹⁾, 311110024@student.machung.ac.id²⁾,
 hendry.setiawan@machung.ac.id³⁾

Abstrak

Pembelajaran di bidang komputasi, terutama yang berfokus pada bidang pengembangan perangkat lunak, membutuhkan ragam keahlian yang mumpuni. Meskipun bidang ilmu komputer menjadi salah satu bidang yang paling digemari oleh para pembelajar dengan kemampuan pendidikan di atas rata-rata, dalam kenyataannya mereka juga memperlihatkan kesulitan pada awal masa tahap pembelajaran. Hal ini dikarenakan karena para pembelajar belum memiliki keahlian khusus dibidang pemrograman komputer. Penelitian ini menghasilkan sebuah purwarupa perangkat lunak yang dapat digunakan untuk pembelajaran dasar logika pemrograman yang menggabungkan konsep permainan dan konsep pembelajaran pemrograman berbasis pola (pattern-based programming). Perangkat lunak yang dikembangkan akan ditujukan untuk media perangkat bergerak berbasis Android untuk tujuan skalabilitas penggunaan. Konsep pembelajaran pemrograman berbasis pola telah dibuktikan mampu meningkatkan motivasi para pembelajar serta kemampuan pemrograman mereka secara spesifik. Penelitian yang sudah dilakukan saat ini hanya memberikan paparan hasil yang diperoleh dari penerapan permainan yang dapat dimainkan secara nyata dengan supervisi para pengajar. Model pembelajaran ini terbatas pada unsur ruang, waktu dan keterlibatan peserta pelajar. Tujuan dari penelitian ini adalah menghasilkan sebuah aplikasi permainan berbasis perangkat bergerak yang mampu mengatasi batasan-batasan yang ada pada penelitian sebelumnya.

Kata kunci: Algoritma, Game Edukasi, Logika, Pemrograman.

Abstract

Learning in computer science, especially whose focused on software development, requires some qualified skills. Although computer science become the most favored areas of learners with above-average educational abilities, but in reality they also show difficulties at the beginning of learning phase. This happened because learners does not have any special skills in computer programming. This research produces a software prototype that can be used for basic learning of programming logic using game concepts and pattern-based programming. The software developed is an Android-based mobile device for scalability purposes. Pattern-based programming learning has been proven to increase learners motivation as well as their specific programming abilities. The research that has been done now only provides exposure the results obtained from application of games that can be played with real teachers supervision. This learning model is limited to elements of space, time and involvement of student participants. The purpose of this study is to produce a mobile device game application that able to overcome restrictions that exist in previous research.

Keywords : Algorithm, Educational Game, Logic, Programming

1. Pendahuluan

Kurangnya ketersediaan atas tenaga ahli terutama di bidang Teknologi informasi telah menarik perhatian banyak pihak baik pemerintah maupun pihak industri. Dalam sebuah situs pemberitaan negara dikemukakan bagaimana Pemerintah Brazil telah mengalokasi anggaran dana yang sangat besar untuk sektor pendidikan, penelitian, dan teknologi mulai tahun 2012-

2014. Sebagian dari dana tersebut digunakan untuk mencetak tenaga ahli dibidang komputer dan teknisi kelas menengah. Hingga saat ini sebagian besar institusi pendidikan yang ada menawarkan model pembelajaran yang menggabungkan konsep pembelajaran praktis seperti di Sekolah Menengah Kejuruan (SMK) dan konsep pembelajaran konvensional seperti di Sekolah Menengah pada umumnya. Konsep pembelajaran ini memiliki banyak keterbatasan dalam penerapannya sehingga tidak sering tujuan pembelajaran yang ingin dicapai malah tidak terpenuhi.

Tujuan utama dari pembelajaran teknis di bidang pemrograman adalah penguasaan yang baik dari pada konsep dasar logika algoritma dan pemrograman komputer itu sendiri. Meskipun bidang ilmu komputer menjadi salah satu bidang yang paling digemari oleh para pembelajar dengan kemampuan pendidikan di atas rata-rata, dalam kenyataannya mereka juga memperlihatkan kesulitan pada awal masa tahap pembelajaran. Hal ini dikarenakan karena para pembelajar belum memiliki keahlian khusus dibidang pemrograman komputer. Kesulitan ini perlu untuk mendapatkan perhatian untuk diatasi sehingga tidak kemudian berdampak pada motivasi belajar para peserta didik ke depannya. Menurunnya motivasi belajar sudah pasti akan berpengaruh terhadap capaian pembelajaran yang rendah.

Untuk dapat memenuhi tujuan tersebut maka dirasa perlu untuk melakukan peningkatan terhadap kualitas pengajaran dan pembelajaran di bidang teknologi informasi terutama pada subjek pengembangan perangkat lunak. Metode pembelajaran dengan teknik konvensional terbukti tidak dapat memotivasi para peserta belajar untuk memahami apa yang sedang dipelajari. Hal ini kemudian berdampak pada capaian pembelajaran yang rendah [1].

Dalam penelitian ini dihasilkan sebuah aplikasi game edukasi yang menerapkan penggabungan konsep pembelajaran pemrograman berbasis pola dengan konsep permainan. Konsep pembelajaran berbasis pola atau juga sering dikenal dengan model pembelajaran/instruksi berorientasi pada pola (*pattern-oriented instruction*) telah terbukti mampu meningkatkan motivasi pembelajar yang secara langsung berdampak pada meningkatnya kemampuan peserta belajar. Hal ini terutama dapat dilihat dalam kemampuan peserta belajar dalam memahami konsep dasar logika algoritma yang menjadi dasar pengetahuan di bidang pengembangan perangkat lunak.

2. Dasar teori

A. Pemrograman Berbasis Pola

Ragam kendala yang ditemui ketika belajar pemrograman cukup bervariasi, mulai dari kesukaran peserta belajar dalam merepresentasikan model pembelajaran yang bersifat abstrak (*abstract model*) hingga kesukaran yang ditemui dalam proses memecahkan masalah. Bahkan untuk beberapa kasus tertentu juga ditemukan bahwa bidang pelajaran tertentu seperti matematika, fisika dan kimia menjadi hal mendasar yang turut mempengaruhi kemampuan peserta belajar dalam proses pembelajaran [2].

Terdapat dua masalah utama yang ditemui ketika belajar tentang pemrograman, yakni masalah yang dihadapi ketika harus mempelajari bahasa pemrograman baru dan belajar bagaimana dapat menyelesaikan sebuah masalah komputasi yang diberikan [3]. Sesuai dengan masalah pertama yang sudah dikemukakan, komponen permasalahan terletak pada proses pembelajaran sintaks dan semantik pada bahasa pemrograman yang digunakan sebagai media pembelajaran. Permasalahan tadi masih belum ditambah dengan ragam bahasa pemrograman yang ada hingga paradigma pemrograman untuk masing-masing bahasa pemrograman tersebut. Namun demikian, permasalahan yang paling mendasar dari permasalahan utama tersebut adalah bagaimana peserta belajar dapat menginterpretasikan permasalahan yang ada, menemukan solusi permasalahan secara logis dan algoritmik, kemudian dapat menuangkannya ke dalam bahasa pemrograman tertentu (*coding*).

Pengalaman yang dimiliki oleh seorang *programmer* dapat membantu dirinya dalam menyelesaikan permasalahan komputasi yang dihadapi; bagaimana menentukan *statements* yang harus digunakan, pemilihan tipe data (*data type*), hingga model penyelesaian yang dipilih untuk menjadi solusi masalah yang dihadapi. Hal yang sama juga dikemukakan oleh Jhonson dan Soloway yang mengemukakan bahwa pengalaman terdahulu dapat digunakan untuk menemukan solusi dari sebuah permasalahan yang berbeda dari sebelumnya [4]. Hal

inilah yang mendasari model pembelajaran berbasis pola. Pola-pola pemrograman ini kemudian akan dihadirkan sebagai kumpulan alternatif solusi terhadap permasalahan yang diberikan. Peserta belajar akan dapat belajar dengan cepat melalui pelatihan praktis yang diberikan.

Pola pemrograman menggambarkan sebuah solusi dari sebuah permasalahan serta menyediakan informasi bagaimana mereka dapat diimplementasikan untuk bisa menyelesaikan sebuah permasalahan tertentu [1]. Pola-pola pemrograman tersebut dapat membantu para peserta belajar dalam proses pembelajaran pemrograman yang mereka lakukan, mulai dari bagaimana menggunakan pola tertentu, bagaimana melakukan kombinasi yang dibutuhkan, hingga membuat sebuah kombinasi pola yang beragam untuk menghasilkan sebuah solusi [5]. Dalam hal ini para pengajar dituntut untuk dapat menciptakan studi kasus-studi kasus yang dapat merangsang proses pembelajaran para peserta belajar.

Konsep pembelajaran pemrograman berbasis pola ini dapat menjadi solusi untuk mengatasi permasalahan komputasi. Konsep pembelajaran ini menghadirkan pola-pola tertentu sebagai blok dasar yang membangun sebuah program aplikasi. Dengan memahami varian pola yang ada, para peserta belajar diharapkan dapat menentukan, menggabungkan hingga menghasilkan pola pemrograman yang sesuai untuk tiap permasalahan yang dihadapi. Beberapa contoh varian pola pemrograman dapat dilihat pada tabel berikut.

Tabel 1. Variasi Pola Pemrograman

Nama Pola	Jenis Pola	Kode (pseudocode)
Pemilihan sederhana	seleksi	<code>if (a > 0) { b = 1; }</code>
Pemilihan Alternatif	seleksi	<code>if (a > 0) { b = 1; } else { b = 0; }</code>
Pemilihan kompleks	seleksi	<code>if (a > 0) { b = 1; } elseif (a == 0) { b = 0; } else { b = -1; }</code>
Pemilihan kumpulan	seleksi	<code>switch (a) { case 0: b = 0; break; case 1: b = 1; break; default: b = -1; }</code>
Perulangan for	perulangan	<code>for (i=0; i<10; i++) { cout << i << " "; }</code>
Perulangan while	perulangan	<code>while (i < 10) { cout << i << " "; i++; }</code>

B. Pemrograman Komputer

Pemrograman komputer adalah proses untuk menuliskan kode-kode perintah komputer dalam bahasa yang dapat dimengerti komputer dan menggunakan tata bahasa maupun sintaks yang dapat diikuti olehnya sehingga komputer dapat melakukan tugas-tugas yang bermanfaat untuk manusia [6]. Dalam pemrograman komputer, *programmer* selalu memerintahkan komputer untuk melakukan sesuatu yang kemudian dijalankan oleh komputer satu per satu secara sekuensial. Kode perintah yang ditulis juga harus tepat dan tidak bersifat ambigu. Pemrograman komputer juga sering diilustrasikan sebagai sebuah cara bagi manusia untuk memberikan perintah kepada mesin untuk melakukan tugasnya, memecahkan masalah, dan untuk menciptakan interaksi antara manusia dan mesin, sehingga mesin dapat berguna bagi manusia [7].

Dalam pemrograman komputer tidak digunakan bahasa manusia, karena pada dasarnya komputer bersifat mati otak, oleh karenanya untuk membuat program komputer, seorang programmer akan menggunakan bahasa yang dimengerti oleh komputer, bahasa-bahasa ini disebut dengan bahasa pemrograman [8].

Komputer pada umumnya menjalankan sebuah program menurut urutan dan struktur tertentu. Struktur dan urutan ini dikenal dengan istilah *control flow*. *Control flow* untuk tiap-tiap bahasa pemrograman umumnya tidak berbeda jauh, hal ini ditujukan agar program yang

dihasilkan dapat menjadi lebih terorganisir, mudah untuk dimodifikasi (penambahan maupun pengurangan), dan lebih modular. Beberapa elemen-elemen dalam konsep *control flow* :

1. Variabel, adalah elemen yang menyimpan suatu nilai atau data, variabel dapat menyimpan nilai dasar seperti angka hingga sebuah objek yang terdiri dari banyak nilai,
2. Perulangan, adalah elemen yang mengulang sekumpulan perintah, contohnya While loop dan For loop,
3. Kondisional, adalah elemen yang menjalankan sekumpulan perintah apabila sebuah kondisi terpenuhi, biasa dituliskan dengan format If Then Else,
4. Input/Output, adalah elemen yang berinteraksi dengan entitas diluar program, misalnya mencetak teks di layar, menerima masukan pengguna, maupun membaca atau menulis *file*,
5. Subrutin (fungsi dan prosedur), adalah elemen yang digunakan untuk mengelompokkan sekumpulan perintah ke dalam sebuah grup, sehingga dapat digunakan lagi dan lagi.

Dalam pemrograman komputer, seorang *programmer* secara sadar maupun tidak sadar akan melakukan langkah-langkah praktis untuk mendapatkan solusi atas permasalahan yang dihadapinya. Langkah-langkah praktis tersebut antara lain, *requirements analysis* atau mendefinisikan permasalahan yang sedang dihadapinya, *design* atau perencanaan, *coding* atau memprogram, *testing* atau uji coba, dan *debugging* atau memperbaiki *bug* (kesalahan logika) pada program yang telah dibuat :

1. *Requirements analysis*, adalah tahap awal dalam proses pemecahan masalah dalam pemrograman komputer. Pada tahap ini, *programmer* menganalisa dan mendefinisikan masalah yang akan dihadapi.
2. *Design*, pada tahap ini *programmer* akan melakukan perencanaan solusi terhadap permasalahan yang akan dihadapi.
3. *Coding*, pada tahap ini *programmer* melakukan pengkodean sesuai dengan solusi yang telah dirancang pada tahap design.
4. *Testing*, pada tahap ini *programmer* melakukan uji coba kode yang telah dihasilkan terhadap hasil yang menjadi ekspektasi.
5. *Debugging*, apabila kode yang dihasilkan tidak memenuhi ekspektasi, tahap *debugging* dilakukan. Pada tahap ini *programmer* akan melakukan evaluasi terhadap kode, menjalankan lagi kode yang sudah ada, dan mencoba mencari kesalahan yang terdapat dalam kode.

C. Serious Game/Game Edukasi

Game adalah sebuah perlombaan secara fisik maupun psikis yang dijalankan menurut suatu aturan yang spesifik dengan tujuan untuk memberikan kesenangan bagi para pelakunya [9]. Sementara *serious game* merupakan permainan yang bertujuan menggunakan kesenangan bermain *game* untuk lebih lanjut memberikan pelatihan, edukasi, maupun manfaat dan tujuan-tujuan lain bagi pelakunya. Dengan kata lain, *serious game* adalah *game* yang didesain untuk menjadi media untuk:

1. Penyaluran dan penguatan terhadap pengetahuan maupun kemampuan praktis
2. Teknik dan konten yang bersifat persuasif yang ditujukan untuk mengubah perilaku sosial maupun personal penggunaanya, termasuk *game* yang ditujukan sebagai media promosi, *marketing*, maupun perekrutan.

3. Game

Game edukasi Ada's Delivery Service memperkenalkan konsep-konsep dasar mengenai pemrograman komputer. Konsep-konsep yang diperkenalkan dapat dibagi menjadi 2 bagian besar, konsep mengenai bagaimana seorang programmer menyelesaikan sebuah masalah yang dihadapinya (*programming practices*) dan konsep mengenai alur eksekusi dari sebuah program komputer secara umum (konsep *control flow*). *Programming practices* bertujuan untuk memberikan pengalaman cara menyelesaikan sebuah masalah melalui serangkaian alur logika algoritma. Sementara konsep *control flow* lebih kepada eksekusi

logika algoritma yang sudah dibangun ke dalam instruksi-instruksi untuk menghasilkan hasil/output sesuai dengan yang diinginkan. Pada tahapan ini para peserta didik perlu untuk memahami bagaimana instruksi-instruksi yang dibangun akan dijalankan. Pemahaman ini akan digunakan pada tiap tahapan pemrograman komputer, terutama pada tahap desain.

1. *Programming practices*

Dalam game edukasi Ada's Delivery Service, konsep mengenai programming practices didapat pada saat para peserta didik mencoba menyelesaikan suatu level/tantangan dengan menyusun algoritma solusi menggunakan bidak-bidak yang disediakan.

- a. *Requirement analysis*, pada game edukasi Ada's Delivery Service, tahapan ini sama dengan ketika pengguna pertama kali memasuki level permainan. Para peserta didik akan menganalisa dan mempelajari terlebih dahulu permasalahan yang diberikan, pola-pola masalah yang dihadapi pada level tersebut, perintah-perintah yang tersedia, dan berapa banyak instruksi yang diperbolehkan untuk menyelesaikannya.
- b. *Design*, pada tahapan ini, peserta didik mulai merencanakan algoritma solusi yang akan digunakan.
- c. *Coding*, pada tahapan ini, pengguna menyusun perintah yang disimbolkan dengan bidak-bidak ke dalam kontainer fungsi yang diinginkan, penyusunan dilakukan sesuai dengan desain yang telah dibuat pada tahap desain.
- d. *Testing*, pada tahapan ini, pengguna menekan tombol Play dan melakukan uji coba terhadap algoritma solusi yang telah dibuat.
- e. *Debugging*, apabila level tidak terpecahkan setelah tahapan *testing*, maka perlu untuk dilakukan evaluasi algoritma solusi yang telah dibuat, mencari *bug* atau kesalahan di dalamnya, menekan tombol Rewind, kemudian menjalankannya kembali setelah algoritma solusi telah diperbaiki.

2. Konsep *Control Flow*

Dalam game Ada's Delivery Service, terdapat beberapa konsep dasar dari *control flow* sebuah program komputer yang diajarkan, yakni konsep sekuensial, konsep mengenai subrutin, dan konsep perulangan.

- a. Konsep sekuensial menjelaskan bahwa setiap instruksi yang disusun akan dieksekusi secara sekuensial. Pada game edukasi ini setiap bidak-bidak perintah yang telah disusun pada kontainer fungsi untuk menyelesaikan masalah yang ada pada setiap level akan dijalankan secara sekuensial,
- b. Konsep subrutin, pada game edukasi Ada's Delivery Service, peserta didik akan diajarkan untuk menggunakan fungsi untuk menyelesaikan solusi. Peserta didik juga diajarkan mengenai bagaimana menggunakan ulang fungsi/subrutin yang telah didefinisikan dan bagaimana mengenali pola pada masalah yang dihadapi kemudian menggunakan fungsi untuk menyelesaikan pola masalah yang ada. Ketika penunjuk perintah yang sedang dijalankan menunjuk referensi pada fungsi, setiap perintah pada fungsi yang ditunjuk akan dijalankan secara berurutan kemudian setelah semua perintah dijalankan, penunjuk akan menjalankan perintah setelah referensi fungsi tersebut. Hal ini sama persis dengan cara kerja bahasa-bahasa pemrograman pada umumnya.
- c. Konsep perulangan, pada game edukasi Ada's Delivery Service memuat 2 macam konsep perulangan yang dapat dilakukan pada game edukasi Ada's Delivery Service. Perulangan rekursi (*infinite looping*) dilakukan dengan cara memanggil sebuah fungsi setelah fungsi tersebut selesai dijalankan. Sedangkan count-controlled looping, dapat dilakukan dengan memanggil sebuah fungsi beberapa kali sesuai dengan count atau jumlah pemanggilan fungsi.

4. Pengujian dan Pembahasan

Pada game edukasi Ada's Delivery Service, antarmuka yang digunakan terbagi menjadi beberapa layar-layar dan dialog-dialog pendukung. Layar-layar yang terdapat pada

game edukasi ini antara lain: layar menu utama, layar permainan, layar tutorial dan layar editor peta.

A. Layar Pemuatan

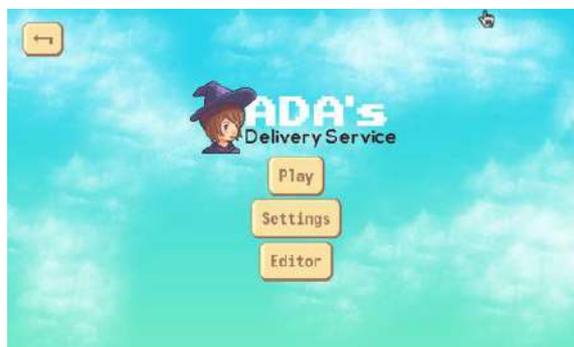
Layar ini adalah layar yang akan dijumpai pengguna ketika aplikasi pertama kali dibuka. Layar ini berfungsi untuk menampilkan progress dari pemuatan game edukasi. Pada layar ini terdapat *progress bar* yang berfungsi untuk menunjukkan *progress* pemuatan aset-aset serta modul-modul *game* yang diperlukan. Aset dan modul yang akan digunakan harus diload terlebih dahulu sehingga tidak mengganggu ketika permainan sedang berlangsung. Ketika proses pemuatan berlangsung, pengguna akan disugahi trivia atau fakta-fakta unik mengenai pemrograman komputer pada bagian tengah layar.



Gambar 1. Antarmuka Layar Pemuatan Ada's Delivery Service

B. Layar Menu Utama

Pada layar ini terdapat 3 tombol utama, yaitu tombol play, tombol setting, dan tombol editor. Tombol play digunakan untuk menuju ke layar pemilihan level, tombol setting digunakan untuk menampilkan dialog pengaturan, sedangkan tombol editor digunakan untuk memulai *editor* peta.



Gambar 2. Antarmuka Layar Menu Utama Ada's Delivery Service

C. Layar Permainan

Layar permainan adalah layar utama yang digunakan untuk memainkan game edukasi ini. Dalam mengelola logika permainan, layar permainan berperan sebagai *view* yang bertugas untuk menampilkan dan menerima masukan dari pengguna dan meneruskannya ke *game controller*. *Game controller* kemudian akan menerima masukan berupa solusi dalam susunan logika algoritma yang telah diekstrak menjadi sebuah *dictionary*. Setiap layar permainan memiliki tingkat kerumitan yang berbeda di setiap level permainannya. Hal ini ditujukan untuk menjaga agar tujuan pembelajaran yang disajikan melalui permainan ini dapat diterima dengan baik level demi level. Pada setiap level permainan inilah, peserta didik akan mendapatkan pengalaman yang mumpuni untuk bisa menghasilkan solusi dari setiap masalah yang diberikan.



Gambar 3. Antarmuka Layar Permainan *Ada's Delivery Service*

D. Layar Tutorial Permainan

Pada beberapa level tertentu terdapat adegan percakapan para karakter dalam game. Percakapan para karakter menjadi elemen cerita dari game edukasi. Percakapan akan muncul apabila kondisi yang dibutuhkan oleh percakapan tersebut telah terpenuhi, misalnya saja pada saat permainan dimulai atau level berhasil diselesaikan. Selain sebagai elemen cerita dari game edukasi *Ada's Delivery Service*, percakapan yang ditampilkan juga sebagai media untuk menyampaikan mekanisme dari game edukasi secara lisan. Ketika percakapan berusaha untuk menerangkan sebuah benda dalam game edukasi, percakapan akan menampilkan sebuah benda secara terfokus sehingga peserta didik dapat lebih memahami setiap aset yang terlibat di dalam permainan di setiap level.



Gambar 4. Antarmuka Layar Tutorial *Ada's Delivery Service*

E. Layar Map Editor

Layar editor peta digunakan untuk membuat, menyimpan, maupun memuat peta selain peta yang sudah disediakan secara default oleh aplikasi. Antarmuka di layar ini dibagi menjadi 3 bagian utama, pada bagian atas terdapat Filebox yang digunakan untuk melakukan segala sesuatu yang berhubungan dengan file peta yang aktif. Pada bagian ini terdapat beberapa tombol, tombol load digunakan untuk memuat peta dari file, kemudian tombol save digunakan untuk menyimpan peta ke dalam *file*, dan tombol play digunakan untuk memainkan peta dalam mode pratinjau. Pada bagian kiri terdapat *toolbar* yang berisi tombol editor atribut peta yang digunakan untuk menyunting atribut level seperti posisi pemain, jumlah fungsi yang diperbolehkan, dan lain-lain, tombol tampilkan atau sembunyikan objek pada peta digunakan untuk menyembunyikan sementara objek atau menampilkannya kembali, fitur ini dibuat agar ketika pengguna membuat peta, pandangan tidak terhalang oleh objek-objek pada peta, dan tombol undo yang digunakan untuk mengulangi perintah yang baru saja dilakukan. Pada bagian kanan layar terdapat *toolbox* yang menampung berbagai macam kuas yang digunakan dalam proses penyuntingan peta, seperti kuas untuk masing-masing petak, kuas untuk menghapus petak dari peta, dan kuas untuk menyunting atribut dari sebuah objek pada peta.



Gambar 5. Antarmuka Layar Map Editor Ada's Delivery Service

5. Kesimpulan

Berdasarkan hasil penelitian yang sudah dilakukan seluruh aset permainan, seperti gambar, suara, efek suara serta modul permainan yang terdapat di dalam aplikasi game edukasi Ada's Delivery Service sudah dapat berjalan dengan baik. Level permainan di desain dengan menerapkan konsep pembelajaran berbasis pola di setiap level permainan untuk memastikan proses pembelajaran yang dilakukan lebih terarah dan jelas. Fitur *map editor* dapat digunakan sebagai media untuk mengasah logika dasar peserta didik dengan model permainan yang lebih kustom diluar materi pembelajaran yang sudah tersaji di setiap level permainannya. Game edukasi Ada's Delivery Service ini dapat digunakan sebagai salah satu alternatif media pembelajaran dasar logika algoritma peserta didik.

Daftar Pustaka

- [1] L.N. Barros, A.P.S. Mota, V.K. Delgado, P.M. Matsumoto, "A Learning Tool for Programming with Pedagogical Patterns", Proceedings of The 005 OOPSLA-Eclipse Workshop of Technology eXchange, ACM Press, pp. 25-129, 2005.
- [2] A.V. de Aquino Leal dan D.J. Ferreira, "Teaching Computer Programming Based on Patterns with Activities and Collaborative Games using Concrete Materials for High School Students", IEEE Frontiers in Education Conference (FIE), pp. 1604-1610, Oktober 2013.
- [3] E. Soloway, "Learning to Program: Learning to Construct Mechanisms and Explanations in: Communications of The ACM", ACM Press, pp. 850-858, 1986.
- [4] W.L. Jhonson dan . Soloway, "PROUST: Knowledge-based Program Understanding", IEEE Transactions on Software Engineering, pp. 369-380, 1985.
- [5] M.J. Clancy dan M.C. Linn, "Patterns and Pedagogy", The Proceedings of The Thirtieth SIGCSE Technical Symposium on Computer Science Education, pp. 37-42, Maret 1999.
- [6] A. Kingsley-Hughes & K. Kingsley-Hughes, "Beginning Programming", New York: John Wiley & Sons, 2005.
- [7] S. Bebbington, "What is Programming", 2014.
- [8] W. Wang, "Beginning Programming for Dummies", New York: John Wiley & Sons, 2011.
- [9] M. Zyda, "From Visual Simulation to Virtual Reality to Games", IEEE Computer Society, pp.25-32, 2005.
- [10] N. Iuppa & T. Borst, "End-to-End Game Development: Creating Independent Serious Game and Simulations form Start to Finish", New York: Taylor & Francis, 2012.2