

Pengembangan Gim Berbasis Web Menggunakan Construct 3 bagi Anak-Anak untuk Belajar Logika Pemrograman

Muhammad Nur Abdurrauf, Imam Kuswardayan, dan Hadziq Fabroyir
Departemen Informatika, Institut Teknologi Sepuluh Nopember (ITS)
e-mail: imam@its.ac.id

Abstrak—Ilmu pemrograman sudah selayaknya menjadi salah satu ilmu dasar bagi anak-anak di era digital ini. Salah satu kemampuan yang menjadi pondasi penting dalam pemrograman adalah pemahaman logika pemrograman. Oleh karena itu, dikembangkanlah sebuah gim edukasi berbasis web untuk mengenalkan dasar-dasar logika pemrograman pada anak-anak secara menyenangkan. Pada gim tersebut, pemain akan menggerakkan sebuah atau beberapa karakter ke portal dengan menyusun blok kode yang telah disediakan untuk menyelesaikan level. Level pada gim dapat memiliki rintangan yang menghalangi pemain mencapai objektifnya. Gim ini memiliki tiga puluh level yang dirancang berdasarkan logika pemrograman yang diajarkan. Setiap level memiliki online leaderboard dan rating bintang untuk mendorong pemain mencari solusi yang optimal dan meningkatkan replayability level. Gim ini berjalan di web browser dan dapat dimainkan di desktop ataupun mobile. Gim akan dikembangkan dengan game engine Construct 3. Selain itu, dikembangkan juga sebuah REST API untuk leaderboard menggunakan framework Hapi. Uji coba gim dibagi menjadi dua bagian, yaitu pengujian fungsional dan pengujian pengguna. Hasil dari pengujian fungsional menunjukkan gim telah berjalan dengan baik sesuai rancangan. Namun, pengujian pengguna menemukan bahwa gim masih terlalu sulit bagi target pemain. Meskipun demikian, gim berhasil membuat pemain belajar logika pemrograman dan merasa senang saat dimainkan.

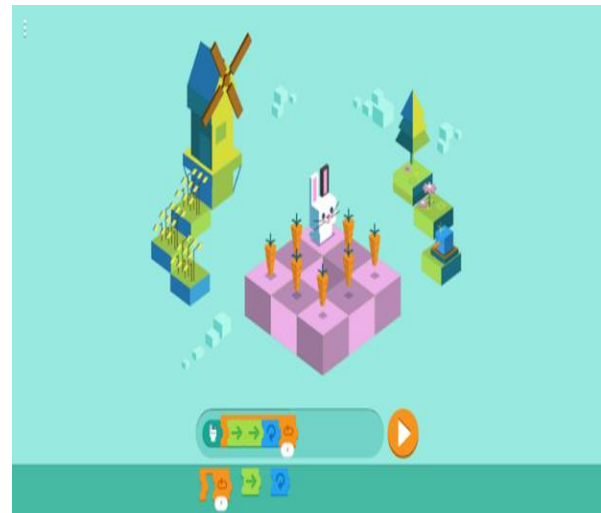
Kata Kunci—Construct 3, Gim Berbasis Web, Gim Edukasi, Hapi, REST API.

I. PENDAHULUAN

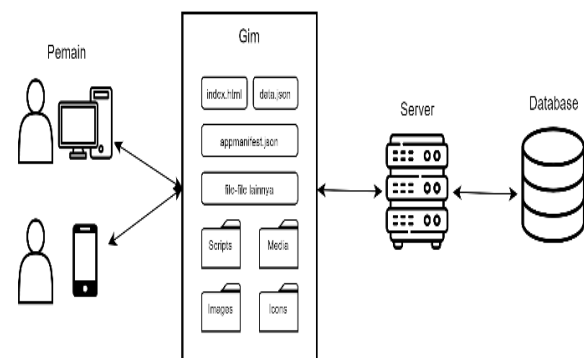
REVOLUSI Industri 4.0 mendorong penggunaan teknologi digital dalam berbagai aspek kehidupan. Ditambah dengan adanya Pandemi Covid-19, transformasi digital ini terjadi semakin cepat. Adanya teknologi-teknologi ini didukung oleh perangkat lunak yang ditulis programmer. Kondisi ini menyebabkan kebutuhan programmer yang handal terus meningkat. Akan tetapi, jumlah programmer yang ada di Indonesia masih kurang jika dibandingkan kebutuhan yang ada. Presiden Joko Widodo bahkan mengatakan bahwa Indonesia membutuhkan setidaknya 9 juta talent digital untuk 15 tahun ke depan.

Programmer sendiri adalah seseorang yang memiliki skill untuk membuat sebuah sistem dengan menggunakan bahasa pemrograman. Bahasa pemrograman tersusun dari beberapa sintaks yang membentuk instruksi yang akan memerintah komputer untuk menjalankan suatu fungsi. Meskipun programming adalah sebuah profesi pekerjaan, tetapi ilmu dalam menulis program ini sudah selayaknya menjadi salah satu ilmu dasar bagi anak-anak di era digital ini.

Menurut sebuah survei di Statista, 68.23% programmer menulis kode pertamanya sebelum mereka berusia tujuh belas tahun. Bahkan 15.17% mengatakan bahwa mereka menulis



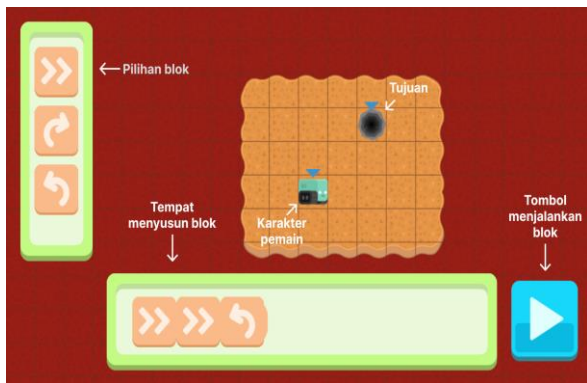
Gambar 1. Tangkapan Layar Coding for Carrots.



Gambar 2. Arsitektur Gim.

kode pertama mereka sebelum sepuluh tahun. Banyak sekali manfaat yang akan didapatkan anak jika belajar pemrograman sejak dini. Mulai dari melatih kemampuan problem solving, computational thinking, meningkatkan kreativitas, hingga meningkatkan komunikasi dan kerja tim mereka.

Salah satu kemampuan yang diperlukan dalam pemrograman adalah pemahaman operasi logika terhadap data ke dalam ilmu komputer atau bisa disebut logika pemrograman. Logika pemrograman ini akan menjadi pondasi yang penting dalam menyusun sebuah algoritma untuk memecahkan berbagai masalah pemrograman sehingga penting untuk dipelajari sedini mungkin. Akan tetapi, mempelajari logika pemrograman ini kurang menarik dan cukup sulit dipahami bagi anak-anak. Oleh karena itu, dikembangkanlah sebuah gim edukasi berbasis web untuk mengenalkan dasar-dasar logika pemrograman pada anak-anak secara menyenangkan untuk Tugas Akhir ini.



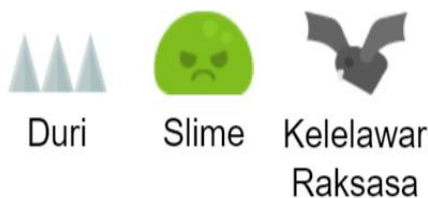
Gambar .3 Tampilan Gim.



Gambar 4. Objek Permata.



Gambar 5. Objek Batu.



Gambar 6. Objek Musuh.

II. TINJAUAN PUSTAKA

A. Logika Pemrograman

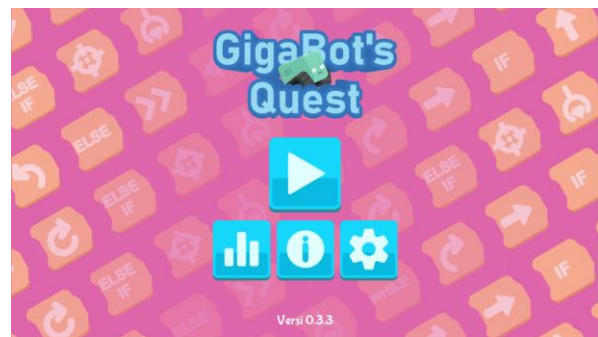
Logika adalah cara berpikir yang didasarkan pada rangkaian prinsip dan aturan tertentu sesuai dengan nalar dan ilmu pengetahuan. Dari pengertian tersebut, logika pemrograman dapat diartikan sebagai sekumpulan prinsip dan aturan yang digunakan dalam menginstruksikan suatu komputer untuk melakukan tugas-tugas tertentu.

1) Logika Kondisional

Logika Kondisional adalah logika yang digunakan untuk memeriksa suatu kondisi dan menjalankan sekumpulan instruksi bila kondisi tersebut terpenuhi. Dengan begitu, komputer dapat melakukan hal yang berbeda untuk kondisi yang berbeda dan membentuk percabangan. Dalam pengecekan kondisi, umumnya digunakan operator perbandingan yang dapat membandingkan dua buah nilai dan menghasilkan nilai *true* atau *false*. Apabila terdapat dua atau lebih kondisi yang perlu diperiksa untuk menjalankan instruksi tertentu, kondisi tersebut dapat digabungkan menjadi satu dengan bantuan operator logika.



Gambar 7. Splash Screen.



Gambar 8. Halaman Menu.



Gambar 9. Pop Up Leaderboard.

2) Logika Perulangan

Logika Perulangan adalah proses pengulangan instruksi yang dilakukan dalam kondisi yang telah ditentukan. Pengulangan dilakukan untuk menghindari penulisan kode yang sama berulang kali, sehingga proses menulis kode menjadi lebih efektif dan efisien.

3) Logika Aritmatika

Logika Aritmatika adalah logika yang menerapkan operasi matematika dalam program. Perhitungan matematika umum digunakan dalam memecahkan suatu masalah dalam pemrograman, sehingga logika ini diperlukan untuk pemecahan masalah tersebut. Untuk melakukan operasi matematika, diperlukan operator untuk memproses operasi tersebut.

B. Edukasi Logika Pemrograman

Konsep logika pemrograman dapat dengan mudah diterapkan pada bahasa pemrograman apapun bila sudah dipahami, bahkan jika hanya dipelajari menggunakan suatu bahasa tertentu. Oleh karena itu, edukasi logika pemrograman hampir selalu terikat dengan pembelajaran dasar-dasar suatu bahasa pemrograman.

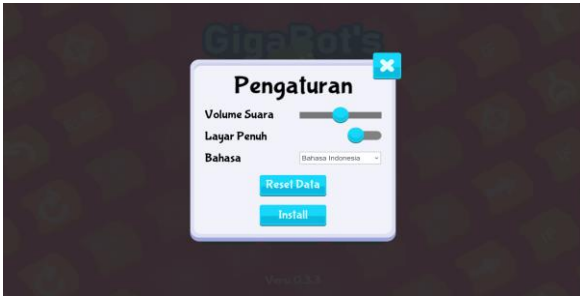
Terdapat juga beberapa gim edukasi yang mengajarkan logika pemrograman, salah satunya Coding for Carrots yang



Gambar 13. Pop Up Credits.



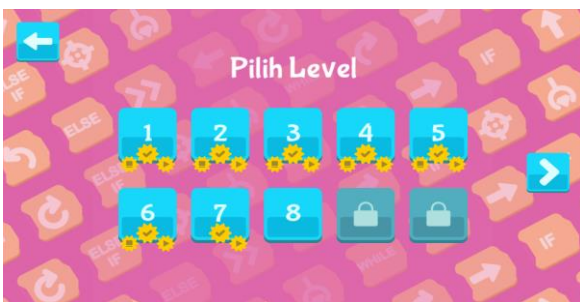
Gambar 10. Pop Up Level Selesai.



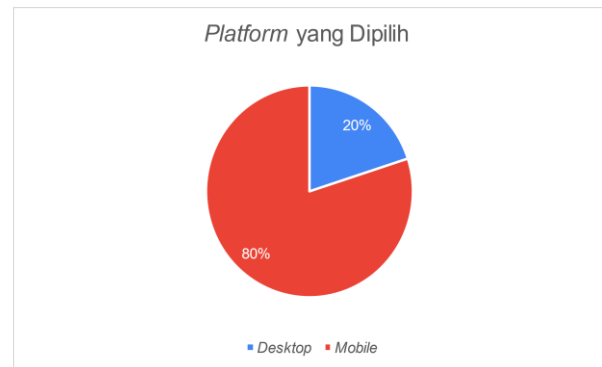
Gambar 14 .Pop Up Pengaturan.

Leaderboard		
id	Integer	M PI
level	Integer	M
username	Varchar(%)	M
steps	Integer	M
code_blocks	Integer	M
time_ms	Integer	M
created_at	Timestamp	M

Gambar 11. Struktur Tabel Leaderboard.



Gambar 15. Halaman Pilih Level.



Gambar 12. Grafik Platform Pilihan Partisipan.



Gambar 16. Halaman Level.

dikembangkan oleh Google. Pada gim tersebut, pemain harus membantu seekor kelinci untuk mengumpulkan wortel. Pemain akan memprogram pergerakan kelinci tersebut dengan blok kode yang didasarkan pada Bahasa Pemrograman Scratch. Tangkapan layar untuk gim tersebut ditunjukkan pada Gambar 1.

C. Gim Berbasis Web

Gim adalah sebuah permainan elektronik yang melibatkan interaksi dengan sebuah antarmuka pengguna ataupun sebuah perangkat untuk menghasilkan umpan balik visual [1]. Gim mempunyai elemen-elemen yang umumnya ada pada permainan secara tradisional, seperti aturan, konflik, dan hasil yang ingin dicapai. Yang membedakan adalah beberapa elemen tersebut akan diatur oleh sebuah komputer.

Sebuah gim bisa dibagi berdasarkan *platform* tempat aplikasi itu berjalan, salah satunya adalah *web browser*. Gim yang dapat berjalan di *browser* itu disebut gim berbasis web.

Tabel 1.

Hasil Pengujian Fungsional Gim

Pengujian	Total Skenario	Hasil
Halaman menu	4	Berhasil
Pop up leaderboard	5	Berhasil
Pop up credits	1	Berhasil
Pop up pengaturan	6	Berhasil
Halaman pilih level	4	Berhasil
Halaman level	11	Berhasil
Pop up level selesai	5	Berhasil

Karena gim berbasis web dijalankan pada *browser*, gim tersebut umumnya tidak perlu di-*install* dan dapat dijalankan di berbagai sistem dan perangkat. Karena hal itu juga, gim berbasis web pada umumnya dirancang untuk tidak memiliki grafis yang intensif dan mempunyai persyaratan minimum sistem yang rendah.

1) Progressive Web App (PWA)

Progressive Web App atau PWA adalah sebuah aplikasi berbasis web yang dikembangkan dengan berbagai teknologi dan digabungkan dengan *progressive enhancement* sehingga memungkinkan aplikasi untuk memanfaatkan fitur-fitur yang terdapat pada aplikasi web dan juga aplikasi *native*. Dengan PWA, aplikasi dapat diakses dengan URL seperti *website* pada umumnya melalui *browser*, tetapi pengguna dapat memilih untuk menginstall aplikasi sehingga bisa terus diakses secara *offline*. Selain itu, PWA juga mendukung fitur-

Tabel 2.
Hasil Penilaian Partisipan

Pernyataan	Penilaian				Rata-Rata
	1	2	3	4	
Mekanika gim mudah dipahami	0	3	5	2	2,9
Tingkat kesulitan gim sudah sesuai dengan kemampuan pemain	0	5	5	0	2,5
Level-level gim dirancang dengan baik	1	4	3	2	2,6
Leaderboard/papan peringkat memotivasi pemain dalam menyelesaikan level	0	3	4	3	3
Rating bintang memotivasi pemain dalam menyelesaikan level	0	3	5	2	2,9
Visual dan grafik dalam gim menarik	0	2	5	3	3,1
Antarmuka pengguna (UI) gim mudah digunakan	0	1	6	3	3,2
Gim berjalan dengan lancar tanpa ada lag, bug, atau crash	0	0	4	6	3,6
Pemain mempelajari hal baru terkait logika pemrograman	0	0	6	4	3,4
Pemain merasa senang ketika bermain gim ini	0	1	5	4	3,3

fitur lain pada aplikasi *native* seperti *push notification* atau akses langsung ke penyimpanan sistem.

2) HTML5

HTML5 adalah versi kelima dan versi major terakhir HTML yang merupakan World Wide Web Consortium (W3C) Recommendation. HTML atau HyperText Markup Language sendiri adalah sebuah *markup language* yang menjadi standar penulisan dokumen yang dibuat untuk ditampilkan di *web browser*. HTML5 bertujuan untuk meningkatkan HTML dengan menambahkan dukungan untuk multimedia dan fitur baru lainnya dengan tambahan *element* seperti video, audio, dan *canvas*.

3) JavaScript

JavaScript adalah sebuah bahasa yang bersifat *interpreted*, *prototype-based*, *multi-paradigm*, *dynamic*, dan mendukung berbagai gaya pemrograman seperti imperatif, deklaratif, ataupun berbasis objek. JavaScript umumnya digunakan pada halaman website untuk memberikan interaktivitas pada sisi klien. Meskipun begitu, JavaScript juga digunakan di banyak *non-browser environment*, seperti pada *server* dengan Node.js atau untuk membuat aplikasi dengan React Native atau Cordova.

D. Construct 3

Construct 3 ada sebuah 2D game engine berbasis web yang mulai dikembangkan di tahun 2017 oleh Scirra Ltd. Cara utama untuk memprogram sebuah gim di Construct 3 adalah menggunakan *visual programming* di *event sheets*. Setiap *event sheet* terdiri dari sekumpulan *event* yang memiliki yang memiliki *condition* dan *action*. Jika *condition* dari sebuah sudah terpenuhi, maka *event* tersebut akan menjalankan *action*-nya.

Dengan penggunaan *event logic* seperti *or* dan *and*, *sub-event*, dan *event group*, Construct 3 memungkinkan pengembangan gim yang canggih tanpa perlu mempelajari bahasa pemrograman yang relatif lebih sulit. Meskipun begitu, Construct 3 juga mendukung JavaScript sebagai bahasa *scripting* tambahan untuk memenuhi kebutuhan penggunaan yang lebih *advance*.

Tabel 3.
Hasil Pengujian Fungsional REST API

Pengujian	Total Skenario	Hasil
Menambahkan data	2	Berhasil
Mengambil data	2	Berhasil

Tabel 4.
Durasi dan Level Partisipan

Kode Partisipan	Durasi	Level Selesai
U01	49	15
U02	33	12
U03	36	13
U04	27	9
U05	70	15
U06	50	14
U07	48	16
U08	189	24
U09	43	16
U10	31	10
Minimum	27	9
Maksimum	189	24
Median	45,5	14,5
Rata-rata	57,6	14,4

Tabel 5.
Kritik dan Saran yang Diterima

Kode Partisipan	Kritik dan Saran
U01	Kurang gampang
U04	Level terlalu susah
U05	Terlalu sulit, lebih bagus dibuat lebih gampang

E. Backend

Backend adalah bagian aplikasi yang bertanggung jawab untuk menyediakan kebutuhan yang tidak terlihat oleh pengguna, seperti bagaimana data disimpan, diolah, juga ditransaksikan secara aman.

1) REST API

API atau Application Program Interface merupakan aturan yang harus diikuti untuk berkomunikasi dengan sistem lain. Sedangkan REST atau Representational State Transfer adalah salah satu arsitektur aplikasi yang memberlakukan suatu syarat tentang bagaimana cara API bekerja. Dari dua definisi tersebut, REST API dapat diartikan sebagai sebuah API yang menerapkan arsitektur REST.

Arsitektur REST API berbentuk *client-server*, artinya *client* mengirimkan sebuah *request* ke *server* sesuai dengan format yang telah ditentukan dan *server* akan memberikan sebuah *response*, baik itu *response* berhasil ataupun gagal. REST API bersifat *stateless*, *server* tidak menyimpan *state* apapun terkait *client*, termasuk *session*. Selain itu, REST API juga independen terhadap teknologi yang digunakan, aplikasi *client* dan *server* dapat dikembangkan menggunakan bahasa pemrograman apapun tanpa mengubah desain API.

2) HAPI

Hapi adalah sebuah *framework open source* untuk Node.js yang dikembangkan oleh Walmart Labs. Hapi dapat digunakan untuk mengembangkan *web server* yang aman dan *scalable*, termasuk REST API. Hapi menyediakan fitur dan *plugin* yang lengkap untuk mengembangkan *web server* yang kompleks, seperti untuk menerapkan *layer authentication*, *tokenize*, *cors*, dan lain sebagainya.

3) PostgreSQL

PostgreSQL atau Postgres adalah sebuah sistem manajemen *database relasional* yang bersifat *open source*. Postgres dapat berjalan di berbagai sistem operasi, termasuk Windows, Linux, atau MacOS. Selain itu, Postgres juga didukung oleh banyak bahasa, seperti C++, PHP, Java, dan JavaScript. Postgres memiliki reputasi yang tinggi terkait arsitektur, keandalan, integritas data, dan ekstensibilitasnya.

III. PERANCANGAN

Gim dikembangkan sebagai PWA, sehingga dapat di-*install* dalam perangkat pemain dan dimainkan secara *offline* meskipun berbasis web selama *browser* yang digunakan mendukung. Gim diimplementasikan dengan menggunakan sistem *event sheet* dan JavaScript di Construct 3. *Event sheet* dan JavaScript bisa saling memanggil fungsi dengan menggunakan API yang telah disediakan Construct 3. Gim juga memiliki *online leaderboard* untuk setiap level di dalamnya. Untuk *leaderboard* tersebut, gim akan melakukan *request* ke sebuah *server* menggunakan REST API yang kemudian akan membaca atau menyimpan data ke sebuah *database*. REST API tersebut diimplementasikan dengan *framework* Hapi, sedangkan *database*-nya menggunakan PostgreSQL. Arsitektur gim ditunjukkan pada Gambar 2.

A. Rancangan Mekanika Gim

Gim yang dikembangkan memiliki perspektif top-down dimana akan ada satu atau lebih karakter pemain yang diletakkan dalam sebuah dunia berbasis petak, contohnya seperti yang ditampilkan pada Gambar 3. Pemain dapat menyelesaikan suatu level dengan menggerakkan karakternya menuju ke sebuah titik akhir yang berupa portal. Karakter tersebut hanya dapat digerakkan dengan menggunakan blok-blok kode yang telah disediakan. Pemain dapat menyusun blok kode tersebut dengan menyeret blok tersebut dan melepaskannya di tempat yang telah disediakan. Sehingga untuk menyelesaikan suatu level, pemain diharuskan untuk menyusun blok kode dengan mengaplikasikan logika pemrograman yang telah dipelajari.

Level dalam gim dirancang berdasarkan konsep logika pemrograman yang diajarkan. Oleh karena itu, blok kode yang disediakan dalam suatu level juga disesuaikan dengan konsep logika pemrograman yang telah diajarkan. Sehingga semakin tinggi level yang dimainkan, semakin banyak variasi blok kode yang tersedia. Blok kode ini sendiri akan dibagi menjadi dua kategori, yaitu blok perintah dan blok ekspresi.

Blok perintah adalah blok yang akan menjalankan suatu perintah pada karakter pemain. Blok ini didasarkan pada konsep *statement* dalam pemrograman. Blok ini dibagi kembali menjadi tiga jenis, yaitu: (1) Blok aksi, terdiri dari blok untuk bergerak, berputar, menembak, dan mengambil permata; (2) Blok percabangan, terdiri dari blok untuk *if*, *else if*, dan *else*; (3) Blok perulangan, terdiri dari blok untuk *for* dan *while*.

Blok ekspresi, yaitu blok yang akan mengembalikan nilai tertentu. Blok ini didasarkan pada konsep *expression* dalam pemrograman dan akan digunakan dalam kondisi pada beberapa Blok Perintah. Blok ini dibagi kembali menjadi empat jenis, yaitu: (1) Blok angka, mengembalikan nilai nol sampai sembilan; (2) Blok operator, mengembalikan operator-

operator aritmatika, perbandingan, dan logika yang umum digunakan; (3) Blok baca, mengembalikan nilai petak di sekeliling karakter pemain; (4) Blok variabel, mengembalikan nilai variabel milik karakter pemain.

Sebuah level akan memiliki rintangan yang akan menghalangi pemain untuk menyelesaikan objektifnya. Rintangan tersebut berupa permata, batu, dan musuh. Permata adalah item yang harus dikumpulkan karakter pemain untuk membuka portal, seperti yang ditampilkan pada Gambar 4. Jumlah permata yang perlu dikumpulkan akan ditampilkan dalam bentuk variabel dan dapat diambil nilainya dengan blok variabel.

Objek batu akan menghalangi jalan robot. Objek ini akan dibagi menjadi dua jenis, yaitu batu biasa dan batu lapuk, seperti yang ditampilkan pada Gambar 5. Pemain dapat menembak batu yang lapuk untuk membuka jalan selama karakter pemain masih memiliki peluru. Jumlah peluru yang dimiliki karakter disimpan dalam sebuah variabel yang dapat digunakan oleh pemain.

Objek musuh dapat merusak zirah karakter apabila terkena karakter pemain. Objek ini akan terdiri dari tiga tipe, yaitu duri, slime, dan kelelawar raksasa, seperti yang ditampilkan pada Gambar 6. Duri dan slime akan bersifat statis di satu petak, sedangkan kelelawar raksasa dapat berpindah petak. Pemain dapat menembak slime dan kelelawar raksasa untuk mengalahkannya, sedangkan duri tidak bisa dihancurkan. Sama seperti jumlah permata dan peluru, nilai batas zirah karakter pemain akan disimpan di sebuah variabel. Apabila nilai zirah karakter nol atau jumlah amunisi karakter nol saat pemain berusaha menembak, maka gim akan menampilkan *error* dan eksekusi blok kode berhenti. Hal ini akan menyimulasikan *runtime error* dalam pemrograman.

Seiring dengan naiknya level dan bertambahnya variasi blok kode yang tersedia, rintangan pada sebuah level akan semakin kompleks sehingga kesulitan level juga akan meningkat. Selain itu, jumlah karakter yang perlu pemain gerakkan dalam satu level juga akan bertambah. Hal tersebut akan berfungsi seperti *test case* untuk mendorong pemain agar mempelajari pola pada level dan menyusun kode blok yang tetap dapat mencapai objektif di kondisi yang sedikit berbeda.

Gim ini sendiri memiliki tiga puluh level. Setiap level memiliki sebuah *rating* bintang dan juga *online leaderboard* masing-masing. Dengan adanya dua hal tersebut, pemain didorong untuk menyelesaikan level sebaik mungkin sehingga dapat meningkatkan *replayability* dari level.

B. Rancangan Antarmuka Gim

Gim ini dirancang untuk dimainkan oleh anak-anak, sehingga antarmukanya perlu dirancang menarik bagi anak supaya dapat meningkatkan minat untuk bermain. Selain itu, gim ini akan dapat dimainkan di *desktop* maupun *mobile*, sehingga antarmuka gim harus dapat mendukung *input* dengan *mouse* dan *keyboard*, juga dengan *touch*.

Gim akan dimulai dari *splash screen* (halaman pembuka), seperti yang dapat dilihat pada Gambar 7. Halaman tersebut akan menampilkan logo-logo dan juga berfungsi sebagai tempat untuk memuat data pemain.

Halaman menu kemudian akan terbuka secara otomatis setelah beberapa detik. Pada halaman tersebut, akan ada pilihan untuk membuka halaman pilih level, *leaderboard*,

credits, dan pengaturan. Tampilan halaman tersebut ditunjukkan pada Gambar 8. Pemain dapat melihat *online leaderboard* dengan membuka *pop up leaderboard*, seperti yang dapat dilihat pada Gambar 9. Pemain dapat memilih level yang ditampilkan dan juga mengubah pengurutan berdasarkan waktu, jumlah langkah, atau jumlah blok kode pada *pop up* tersebut. *Pop up credits* akan menampilkan atribusi untuk semua *asset* yang digunakan, seperti yang ditunjukkan pada Gambar 10.

Pemain dapat mengubah pengaturan gim seperti volume suara dan bahasa pada *pop up* pengaturan, seperti yang dapat dilihat pada Gambar 11. Selain itu terdapat juga opsi untuk mereset data dan meng-*install* gim bila *browser* mendukung.

Halaman pilih level menampilkan semua level yang tersedia, seperti yang ditunjukkan Gambar 12. Halaman tersebut juga menampilkan *rating* bintang untuk setiap level. Pemain dapat memilih level yang ingin dimainkan dengan mengklik salah satu level. Jika level tersebut sudah terbuka, maka gim akan menampilkan halaman untuk level tersebut.

Pada halaman level, terdapat sekumpulan blok kode yang tersedia dan tempat menyusunnya. Seperti yang telah dijelaskan sebelumnya, pemain dapat menyusun blok kode tersebut dengan menyeret dan melepaskan blok tersebut ke tempat yang telah disediakan. Kemudian pemain dapat menekan tombol play untuk menjalankan perintah yang telah dipilih. Pemain juga dapat mendapatkan penjelasan mengenai blok kode dengan menyeret blok ke objek tanda tanya ataupun menyeret objek tanda tanya ke blok kode. Untuk blok perintah percabangan dan perulangan, pemain dapat membuka *pop up* tempat menyusun kondisi dengan mengklik kondisinya pada blok. Selain itu, akan ditampilkan juga level saat ini, waktu berlalu, jumlah kode blok yang digunakan, jumlah aksi, variabel karakter pemain, dan tombol untuk menunda permainan. Tampilan halaman tersebut ditunjukkan pada Gambar 13.

Setelah pemain berhasil menyelesaikan sebuah level, gim akan menampilkan sebuah *pop up* level selesai seperti yang ditunjukkan pada Gambar 14. *Pop up* tersebut akan menampilkan waktu penyelesaian, jumlah kode blok yang digunakan, dan jumlah langkah. Pemain dapat memilih untuk menambahkan data mereka ke *leaderboard* atau tidak. Selain itu, terdapat juga tombol *restart*, tombol ke level selanjutnya, dan tombol untuk kembali ke menu.

C. Rancangan REST API

REST API digunakan dalam implementasi *online leaderboard*. Untuk itu, *database* yang digunakan oleh REST API hanya memiliki satu tabel. Gambar 15 menyimpan level yang diselesaikan, nama pemain, jumlah langkah, jumlah kode blok yang digunakan, lama waktu penyelesaian dalam milidetik, dan kapan data tersebut ditambahkan. Selain itu, terdapat juga id yang akan di-*generate* secara otomatis dan berperan sebagai *primary key*. Struktur tabel tersebut ditunjukkan pada Gambar 15.

REST API memiliki dua buah *endpoint*. Untuk menambahkan data, terdapat *endpoint* dengan *method* POST dan *request body*-nya memiliki data yang sesuai dengan kolom pada tabel kecuali *id* dan *created_at* yang di-*generate* otomatis. Sedangkan untuk mengambil *leaderboard*, terdapat *endpoint* dengan *method* GET dan memiliki *query parameter* *level*, *page*, *pageSize*, *sortBy*, dan *order*.

IV. UJI COBA DAN EVALUASI

A. Pengujian Fungsional

Pengujian fungsional dilakukan untuk memeriksa apakah gim sudah berfungsi sesuai dengan rancangannya. Pengujian ini dilakukan dengan metode *black box*, dimana pengujian dilakukan dengan mengamati masukan dan keluaran dari skenario yang telah ditetapkan tanpa perlu mengetahui struktur kode.

Pengujian fungsional gim dilakukan secara manual oleh penulis. Penulis akan menjalankan setiap skenario, mencatat hasil akhirnya dan membandingkannya dengan hasil yang diharapkan. Hasil dari pengujian fungsional untuk gim ditampilkan pada Tabel 1.

Pengujian fungsional gim ini sendiri dilakukan pada perangkat yang sama dengan pengembangan. Akan tetapi, gim tetap akan memiliki performa yang baik selama dijalankan dengan perangkat yang memiliki *browser* berbasis Chromium dan memiliki layar dengan *aspect ratio* 16:9.

Pengujian fungsional untuk REST API dilakukan dengan bantuan Postman. Penulis akan menggunakan fitur *collection runner* pada Postman yang akan melakukan request dan mengecek response-nya secara otomatis. Hasil dari pengujian tersebut ditampilkan pada Tabel 1.

B. Pengujian Pengguna

Pengujian pengguna dilakukan untuk mengetahui apakah gim yang dikembangkan sudah sesuai dengan target pengguna, yaitu anak-anak sekolah dasar. Untuk Tugas Akhir ini, gim diujikan ke sepuluh anak. Partisipan dibebaskan untuk memilih bermain di *desktop* atau *mobile*. Persebaran data *platform* yang dipilih partisipan ditunjukkan pada Gambar 16. Partisipan juga dibebaskan untuk berhenti kapan pun mereka mau. Data durasi dalam menit dan jumlah level yang diselesaikan untuk setiap partisipan ditunjukkan pada Tabel 2.

Setelah bermain, partisipan diminta untuk memberikan nilai 1-4 untuk pernyataan yang telah disiapkan. Pada skala tersebut, angka 1 mengindikasikan “sangat tidak setuju” dan angka 4 mengindikasikan “sangat setuju”. Kemudian untuk setiap pernyataan akan dihitung total nilai dari semua partisipan dan dibagi jumlah partisipan sehingga didapatkan rata-rata nilai. Hasil penilaian tersebut ditampilkan pada Tabel 3.

Partisipan juga diminta untuk memberi kritik ataupun saran untuk gim. Karena bersifat opsional, tidak semua partisipan memberikan jawaban. Data kritik dan saran dari partisipan ditunjukkan pada Tabel 4.

Dari hasil penilaian pengguna, pernyataan “Tingkat kesulitan gim sudah sesuai dengan kemampuan pemain” dan “Level-level gim dirancang dengan baik” memiliki rata-rata nilai yang cukup rendah, yaitu 2,5 dan 2,6. Kedua nilai ini mengindikasikan bahwa tingkat kesulitan gim dan rancangan level masih belum sepenuhnya sesuai dengan target pemain.

Temuan tersebut didukung oleh data level yang dicapai partisipan. Data tersebut memiliki median 14,5, yang berarti setengah dari partisipan tidak dapat mencapai level 15 dari tiga puluh level yang tersedia pada gim sebelum akhirnya berhenti. Semua kritik dan saran yang diterima dari partisipan juga menyampaikan bahwa gim terlalu sulit.

Ada beberapa kemungkinan penyebab hal tersebut. Penjelasan untuk blok kode dan logika pemrograman

mungkin tidak cukup jelas atau kurang terperinci, sehingga pemain kesulitan memahaminya. Selain itu, jarak antara level yang mengenalkan suatu konsep logika pemrograman juga dapat menjadi salah satu faktor. Jika jarak antara level tersebut terlalu dekat, pemain mungkin tidak memiliki banyak kesempatan untuk melatih dan menguasai suatu konsep sebelum mereka dikenalkan ke konsep lain.

Selain kedua pernyataan yang telah disebutkan sebelumnya, dapat dilihat bahwa pernyataan lainnya memiliki rata-rata nilai yang cukup tinggi dan mayoritas partisipan menjawab positif (setuju atau sangat setuju). Hal ini menunjukkan bahwa selain dari tingkat kesulitan dan rancangan level, gim sudah sesuai dengan target pemain.

V. KESIMPULAN DAN SARAN

Berdasarkan dari hasil pengerjaan dan uji coba yang telah dilakukan, beberapa kesimpulan yang dapat diambil adalah:(1)Gim dirancang untuk mendorong pemain menggunakan logika pemrograman dimana pemain akan menggerakkan satu atau lebih karakter menuju sebuah portal dengan cara menyusun blok kode yang tersedia. Sehingga untuk menyelesaikan level, pemain harus memahami dan mampu mengaplikasikan logika pemrograman yang diajarkan. Dengan begitu, gim telah berhasil mengintegrasikan edukasi logika pemrograman ke dalam mekanikanya;(2)Level dalam gim dirancang dengan memerhatikan logika pemrograman yang akan diajarkan. Setiap level juga dapat memiliki rintangan yang berupa permata, batu, dan musuh. Semakin tinggi level, semakin banyak variasi kode blok dan semakin kompleks juga rintangannya. Sebanyak tiga puluh level telah berhasil dirancang untuk gim ini;(3)Gim telah berhasil

diimplementasikan di Construct 3 dengan menggunakan sistem *event sheet* dan JavaScript. Event di *event sheet* dan kode JavaScript bisa saling berkomunikasi menggunakan API yang disediakan Construct 3. Selain itu, digunakan juga library *mathjs* dalam implementasi blok kode dan *il8next* untuk pengaturan bahasa gim;(4)Uji coba pada gim dibagi menjadi pengujian fungsional untuk mengecek apakah gim sudah sesuai rancangan dan pengujian pengguna untuk memastikan apakah gim sudah sesuai dengan target pengguna. Hasil dari pengujian fungsional menunjukkan bahwa gim sudah berjalan dengan baik sesuai rancangan. Sedangkan hasil dari pengujian pengguna menemukan bahwa gim yang telah dikembangkan masih terlalu sulit untuk target pemain, tetapi berhasil membuat pemain belajar logika pemrograman dan merasa senang saat dimainkan.

Terdapat juga beberapa saran yang didapat dari hasil pengerjaan dan uji coba. Berikut adalah beberapa saran tersebut:(1)Memperinci penjelasan blok kode dan logika pemrograman supaya lebih mudah dipahami pemain;(2)Menambahkan jumlah level sehingga pemain memiliki lebih banyak kesempatan untuk melatih pemahaman logika pemrograman mereka;(3)Mengajarkan konsep pemrograman lain seperti fungsi, tipe dan struktur data, kelas dan objek, dan sebagainya;(4)Membuat *procedural level generator* sehingga level bisa dibuat adaptif terhadap kemampuan pemain.

DAFTAR PUSTAKA

- [1] M. J. P. Wolf, *The Medium of the Video Game*. Texas : University of Texas Press, 2022.