

PERBANDINGAN GAME FRAMEWORK PADA HTML5

¹Moch. Rizki Romdoni, ²Noviana

Sekolah Tinggi Teknologi Indonesia, Jurusan Teknik Informatika
e-mail : ¹rizki@sttindonesia.ac.id, ²viana201@gmail.com

ABSTRAK

Game merupakan aplikasi yang mungkin diminati berbagai kalangan dalam mengisi waktu kosong. Seiring dengan pengembangan game, jumlah orang yang tertarik dalam membuat game juga meningkat. HTML5 merupakan salah satu bahasa pemrograman yang bisa digunakan dalam membuat suatu game. Banyak game framework yang tersedia pada HTML5 yang memungkinkan pengguna kesulitan dalam memilih. Dikarenakan belum adanya penelitian yang membahas tentang perbandingan game framework HTML5, maka penulis mencoba untuk membandingkannya.

Perbandingan game framework meliputi ketersediaan dokumentasi, forum dan performance. Perbandingan dilakukan dengan mengunduh game yang bertema sama dari masing-masing game framework dan kemudian mengukur performance dengan menggunakan Chrome Dev Tools dan Mozilla Dev Tools.

Pada hasil akhir penelitian ini menunjukkan bahwa game framework yang memiliki performance terbaik adalah PixiJS dibandingkan dengan game framework PhaserJS, MelonJS dan KiwiJS.

Kata kunci : *Game Framework, PhaserJS, PixiJS, MelonJS, KiwiJS.*

1. PENDAHULUAN

Game merupakan aplikasi pendukung yang mungkin diminati dalam mengisi waktu kosong maupun untuk mengasah kemampuan otak. Banyak juga kalangan-kalangan mulai dari anak balita sampai dengan orang tua sekali pun senang memainkan game. Seiring dengan perkembangan game yang semakin marak maka, jumlah peminat juga meningkat. Selain jumlah pemain yang tinggi, sekarang ini sudah tidak jarang ditemukan banyak orang yang tertarik dalam membuat atau mengembangkan suatu game.

HTML5 biasanya digunakan untuk membuat dan menampilkan isi dari suatu web. Banyak orang belum mengetahui bahwa HTML5 bisa dimanfaatkan sebagai game development tool. Pada bahasa pemrograman ini tidak membutuhkan aplikasi khusus karena hanya menggunakan game framework. Sebagai web development tool HTML5 tidak sulit untuk kita temukan dokumentasi ataupun kajiannya. Akan tetapi, untuk pembahasan HTML5 sebagai game development tool bisa dikatakan sulit untuk menemukan pembahasannya. Ada juga beberapa artikel yang membahas tentang HTML5 dan game frameworknya. Namun artikel yang tersedia masih belum cukup lengkap dan mendalam.

Dengan berkembangnya HTML5 sebagai game development tool, banyak game framework yang dikembangkan mulai dari yang gratis sampai dengan

yang berbayar. Dimana pada saat memulai pembuatan *game* pada *HTML5* pastinya akan mengalami kesulitan karena harus memilih *game framework* yang akan digunakan dan banyak waktu yang akan terbuang karena pada akhirnya *programmer* hanya akan memilih satu *game framework* saja untuk digunakan. .

Untuk menentukan *game framework* yang cocok dan layak digunakan dalam mempercepat proses pembuatan *game*, dapat ditentukan dari bagaimana kemudahan dalam pembelajaran dan pengembangan pada *user interfacenya*. Kemudian fungsi apa saja dari *game framework*. Poin penting lainnya adalah *game framework* yang dipakai memiliki dokumentasi yang dapat dipelajari dan memiliki komunitas yang aktif. Dikarenakan banyak sekali poin yang perlu diperhatikan dalam memilih *game framework* maka diperlukan adanya suatu cara yang mampu memperkenalkan cara kerja *gameframework*. Oleh karena itu, penulis mengajukan “Perbandingan *GameFramework* pada *HTML5*” sebagai judul skripsi.

2. KAJIAN PUSTAKA

2.1 Definisi Perbandingan

Dalam Kamus Lengkap Bahasa Indonesia disebutkan bahwa perbandingan berasal dari kata banding yang berarti persamaan, selanjutnya membandingkan mempunyai arti mengadu dua hal untuk diketahui perbandingannya. Perbandingan diartikan sebagai selisih persamaan (Bambang Marhiyanto; 57).

Menurut Sjachran Basah (1994: 7), perbandingan merupakan suatu metode pengkajian atau penyelidikan dengan mengadakan perbandingan di antara dua objek kajian atau lebih untuk menambah dan memperdalam pengetahuan tentang objek yang dikaji. Jadi di dalam perbandingan ini terdapat objek yang hendak diperbandingkan yang sudah diketahui sebelumnya, akan tetapi pengetahuan ini belum tegas dan jelas.

Dari pengertian perbandingan diatas dapat disimpulkan bahwa pengertian perbandingan adalah suatu proses untuk membandingkan dua hal atau lebih untuk mengetahui persamaan dan perbedaanya.

2.2 Definisi Game

Game berasal dari kata bahasa inggris yang memiliki arti dasar “Permainan”. Permainan dalam hal ini merujuk pada pengertian “kelincahan intelektual” (*intellectual playability*).

Games are a type of play activity, contacted in the context of a pretended reality, in which the participant(s) try to achieve at least one arbitrary, nontrivial goal by acting in accordance with rules. *Game* adalah salah satu jenis aktivitas bermain, yang didalamnya dilakukan dalam konteks berpura – pura namun terlihat seperti nyata, yang mana pemainnya memiliki tujuan untuk mendapatkan satu kemenangan serta dilakukan dengan sesuai dengan aturan permainan yang dibuat.

2.2.1 Tahapan Dalam Membuat Game

Menurut Samuel Henry dalam bukunya yang berjudul "Panduan Praktis Membuat *Game 3D*" mengatakan bahwa untuk membuat sebuah game harus melewati beberapa tahapan, secara sederhana tahapan membuat game adalah sebagai berikut: Skenario, Tema/Tujuan, Aturan/Rules, Konflik.

2.2.2 Elemen *Game*

Menurut R. D. Duke (1980), sebagaimana dalam Wachowicz, et al. Terdapat 11 elemen game yang perlu diperhatikan sebagai dasar dalam membuat game yang baik, yaitu: *Format, Rules, Policy, Scenario, Events, Roles, Decisions, Levels, Score Model, Indicators, Symbols*.

2.2.3 Jenis-Jenis *Game*

Jenis *game* biasa disebut dengan istilah *genre game*. Selain berarti jenis, *genre* juga berarti gaya atau *format* dari sebuah *game*. Menurut Henry dalam bukunya yang berjudul "Panduan Praktis Membuat *Game 3D*" mengatakan *format* sebuah *game* bisa murni sebuah *genre* atau bisa merupakan campuran (*hybrid*) dari beberapa *genre* lain. Jenis-jenis *game* yang ada menurut Henry adalah sebagai berikut: *Maze Game, Board Game, Card Game, Battle Card Game, Quiz Game, Puzzle Game, Shoot Them Up, Side Scroller Game, Fighting Game, Racing Game, Turn-Based Strategy, Real-Time Strategy Game, Simulation Video Game, First Person Shooter, First Person Shooter 3D, Third person 3D Games, Role Playing Game, Adventure Game, Educational and Edutainment, Game sports*.

Berdasarkan jenis *platform*-nya *game* dikategorikan menjadi: *Arcade, Console, Computer Games, Online Games, Handheld Games, Mobile Games, Tabletop*.

2.3 *Framework*

Framework adalah kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *script* (terutama *class* dan *function*) yang dapat membantu *developer/programmer* dalam menangani berbagai masalah-masalah dalam pemrograman seperti koneksi ke *database*, pemanggilan variabel, *file* dan lain-lain.

Framework mirip dengan *application programming interface* (API), walaupun secara teknik API sudah termasuk dalam *framework*. Seperti namanya *framework* berfungsi sebagai suatu fondasi untuk pemrograman, sementara API menyajikan akses ke elemen-elemen yang didukung oleh *framework*. *Framework* juga meliputi *code libraries, compiler*, dan program lainnya yang digunakan pada proses pengembangan perangkat lunak.

Ada beberapa hal yang harus di pertimbangkan dalam memilih *framework*: Pertama adalah dokumentasi dari *framework* itu sendiri. Berikutnya kesesuaian *framework* tersebut dengan aplikasi apa yang hendak kita kembangkan. Selanjutnya adalah memperhatikan standar-standar *framework* yang ada. Semakin banyak *ibid* yang di *support* oleh *framework* tersebut semakin baik

2.4 *Game Framework*

Game framework is not a game engine. A game engine is more like a finished piece of code that you would use as a black box to make a game; It's more about instantiation and configuration than software engineering. An engine has functions that you use, so you don't have to write the code yourself. A game framework, on the other hand, is used to help you organized your code and game functions. It can start out fairly but can be expanded and changed as your games get more complicated.

Game framework bukan game engine. Game engine terlihat seperti kode yang sudah lengkap dan akan digunakan sebagai black box untuk membuat game; sedangkan game framework lebih ke instansiansi dan konfigurasi dibandingkan dengan software engineering (rekayasa perangkat lunak). Game framework pada pengertian lainnya adalah sesuatu yang digunakan untuk mebantuu dalam mengatur/mengorganisir kode dan function pada game. Dimana game framework bisa dimulai dengan biasa saja tapi dapat dikembangkan dan diubah seiring dengan tingkat kerumitan pada game.

2.5 Perangkat Lunak Yang Digunakan

2.5.1 HTML5

Hyper Text Mark-up Language (HTML) adalah bahasa yang digunakan untuk membuat suatu halaman web dan menampilkan berbagai informasi di dalam sebuah browser internet. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan secara luas untuk menampilkan SGML (Standard Generalized Markup Language), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh World Wide Consortium (W3C).

When you add an X in front of HTML, you get XHTML, a reworked version of HTML based on the eXtensible Markup Language (XML). XML was designed to work and behave well with computers, software, and the Internet. Bila menambahkan X di depan HTML, maka akan mendapatkan XHTML, yaitu sebuah versi yang dikerjakan ulang oleh HTML berdasarkan eXtensible Markup Language (XML). XML dirancang untuk dapat bekerja baik terhadap komputer, perangkat lunak, dan internet.

HTML5 already appears to be succeeding where XHTML did not. Even though the standard is still under construction, HTML5 is widely adopted and used on the web today. HTML5 sudah tampak sukses yang mana sebelumnya tidak pada XHTML. Meskipun standarnya masih dalam proses pengembangan, HTML5 banyak digunakan dan digunakan di web saat ini.

2.5.2 PhaserJS

Phaser is a fast, free, and fun open source HTML5 game framework. It uses a custom build of Pixi.js for WebGL and Canvas rendering, and supports desktop and mobile web browsers. Games can be compiled to iOS, Android and native desktop apps via 3rd party tools.

Phaser merupakan *game framework open source* pada *HTML5* yang cepat, gratis dan menyenangkan. *Phaser* menggunakan *custom build* dari *PixiJS* untuk *WebGL* dan *Canvas rendering*, dan mendukung penggunaan pada *desktop* maupun *web browsers*. *Games* dapat dikompilasi ke *iOS*, *Android* dan aplikasi *desktop* melalui *tools* pihak ke-3.

2.5.3 *PixiJS*

PixiJS is a rendering library that will allow you to create rich, interactive graphics, cross platform applications, and games without having to dive into the WebGL API or deal with browser and device compatibility. PixiJS has full WebGL support and seamlessly falls back to HTML5's canvas if needed.

PixiJS adalah *rendering library* yang memungkinkan untuk membuat grafik yang interaktif, aplikasi multi *platform*, dan permainan tanpa harus mengerti tentang *WebGL API* atau menangani kompatibilitas *browser* dan perangkat. *PixiJS* memiliki dukungan penuh terhadap *WebGL* dan dengan cepat kembali ke kanvas *HTML5* jika diperlukan.

2.5.4 *MelonJS*

melonJS is an open source HTML5 game engine that empowers developers and designers to focus on content. The framework provides a collection of composable entities and support for a number of third-party tools.

melonJS merupakan *open source* dari *game engine HTML5* yang memberikan *developers* dan desainer untuk fokus pada isinya (*content*). *Framework* ini menyediakan berbagai entitas yang dapat diubah entitasnya dan mendukung *tools* dari pihak ketiga.

2.5.5 *KiwiJS*

Kiwi.js is the worlds easiest to use Open Source HTML5 game framework for making both mobile and desktop HTML5 browser games. Our focus is blazingly fast WebGL rendering and complimentary tools to make professional quality serious games. We use CocoonJS for publishing games and App creation.

KiwiJS merupakan *game framework* termudah di dunia untuk digunakan sebagai *game framework HTML5* yang bersifat *open source* untuk membuat *game browser* pada *mobile* maupun *desktop*. *GameFramework* ini berfokus pada *rendering WebGL* yang cepat serta menyediakan *tools* yang gratis untuk menciptakan *game* yang berkualitas dan profesional. *KiwiJS* menggunakan *CocoonJS* untuk menerbitkan *game* dan pembuatan *App*.

2.5.6 *XAMPP*

XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis *PHP* dan menggunakan pengolah data *MySQL* di komputer lokal. *XAMPP* berperan sebagai *server web* pada komputer anda. Nama *XAMPP* merupakan singkatan dari *X* (empat sistem operasi), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia di bawah *GNU General Public License* dan bebas, adalah mudah untuk menggunakan *web server* yang dapat melayani tampilan halaman *web* yang dinamis.

2.5.7 Adobe Photoshop CS3

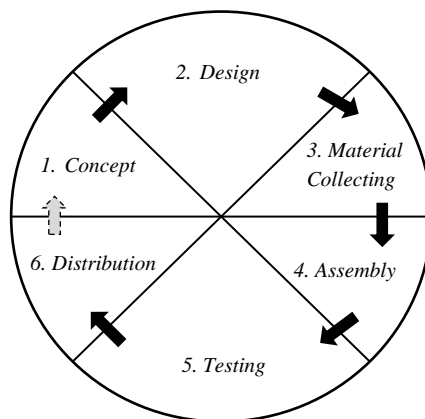
Adobe Photoshop adalah perangkat lunak editor citra buatan *Adobe Systems* yang dikhususkan untuk pengeditan foto/gambar dan pembuatan efek. Perangkat lunak ini banyak digunakan oleh fotografer digital dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (*market leader*) untuk perangkat lunak pengolah gambar/foto, dan, bersama *Adobe Acrobat*, dianggap sebagai produk terbaik yang pernah diproduksi oleh *Adobe Systems*.

2.5.8 Microsoft Office Visio 2007

Microsoft visio (atau sering disebut *visio*) adalah sebuah program aplikasi computer yang sering digunakan untuk membuat diagram, diagram aliran (*flowchart*), *brainstorm*, dan skema jaringan yang dirilis oleh *Microsoft Corporation*. Aplikasi ini menggunakan grafik *vector* untuk membuat diagram-diagramnya.

3. METODOLOGI PENELITIAN

Metode pengembangan sistem yang digunakan pada penelitian ini adalah metodologi *Multimedia Development Life Cycle (MDLC)*. Menurut Sutopo (2003), yang berpendapat bahwa metode pengembangan multimedia terdiri dari 6 tahapan, yaitu *concept*, *design*, *material collecting*, *assembly*, *testing*, *distribution* seperti gambar dibawah ini:



Gambar 1.2 *Multimedia Development Life Cycle (MDLC)*

Adapun tahapan-tahapan pengembangan multimedia pada saat membuat contoh *game* yang diuji sebagai berikut:

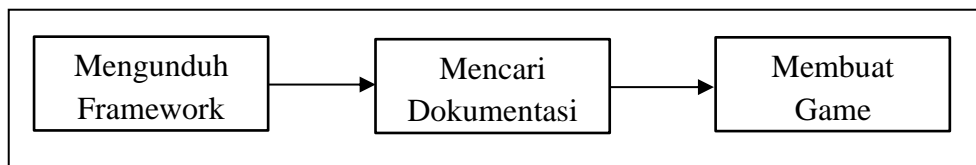
1. *Concept*, dalam tahapan ini, kegiatan yang dilakukan yaitu:
 1. Menentukan jenis *game* yang akan dibuat yaitu *game casual*.
 2. Menentukan jumlah *player* pada *game*.
 3. Menentukan alur dari suatu *game*.
2. *Design*, pada tahap ini, hal yang akan dikerjakan meliputi:
 1. Pembuatan *flowchart game*.
 2. Menentukan gambar atau asset apa saja yang diperlukan dalam *game*.

3. *Material Collecting*, Pengumpulan materi adalah tahap pengumpulan bahan yang sesuai dengan kebutuhan yang dikerjakan. Kemudian yang dilakukan dalam tahap ini adalah mencari dokumentasi – dokumentasi *game framework*.
4. *Assembly*, Tahap *assembly* adalah adalah tahap pembuatan semua objek atau bahan multimedia. Pembuatan aplikasi didasarkan pada tahap *design*. Dalam tahapan ini membuat *game* dengan menggunakan *game framework*.
5. *Testing*, Tahap pengujian setelah menyelesaikan tahap pembuatan (*assembly*) dengan menjalankan aplikasi dan melihat apakah ada kesalahan atau tidak. Pada tahap ini akan menguji apakah *game* dapat dijalankan sesuai dengan *game framework* yang dibuat.
6. *Distribution*, Pada tahap ini aplikasi akan disimpan dalam suatu media penyimpanan. Tahap ini dilakukan jika *game* yang dibuat sudah lengkap yaitu dapat didistribusikan pada *google play store*.

4. HASIL DAN PEMBAHASAN

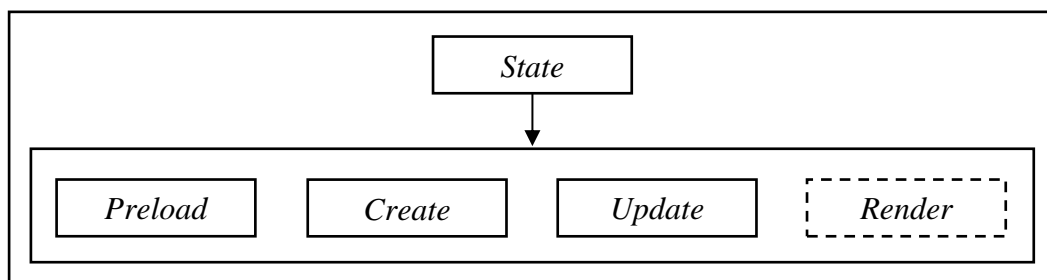
Analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk dan mengevaluasi dan mengetahui bagaimana asal usul sistem yang akan diteliti dan dibangun. Perancangan *game* dengan beberapa *game framework* sebagai berikut:

4.1 *Storyline* dalam merancang sistem



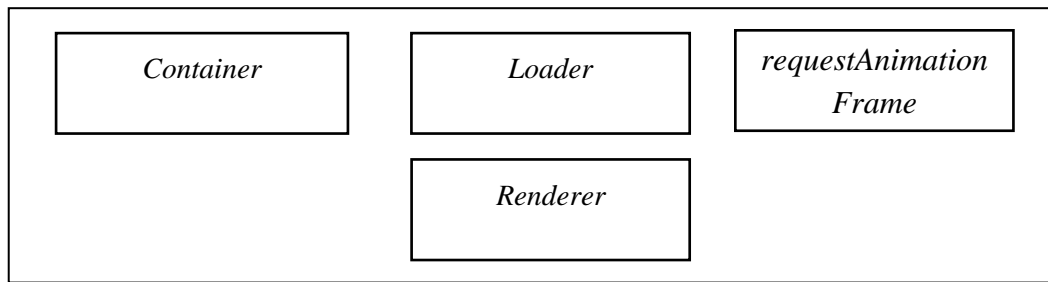
Gambar 1 *Storyline* dalam merancang system

4.2 Tahap merancang *Game* dengan *PhaserJS*



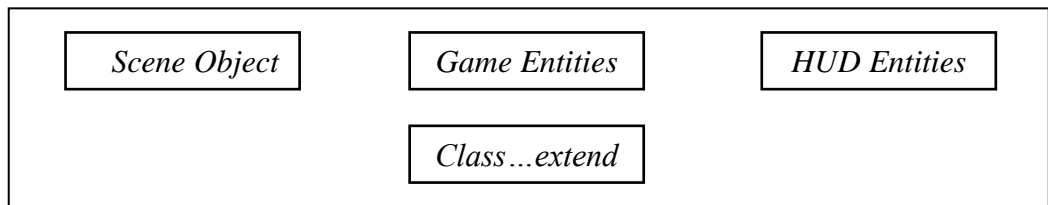
Gambar 2 Tahap merancang *Game* dengan *PhaserJS*

4.3 Tahap merancang *Game* dengan *PixiJS*



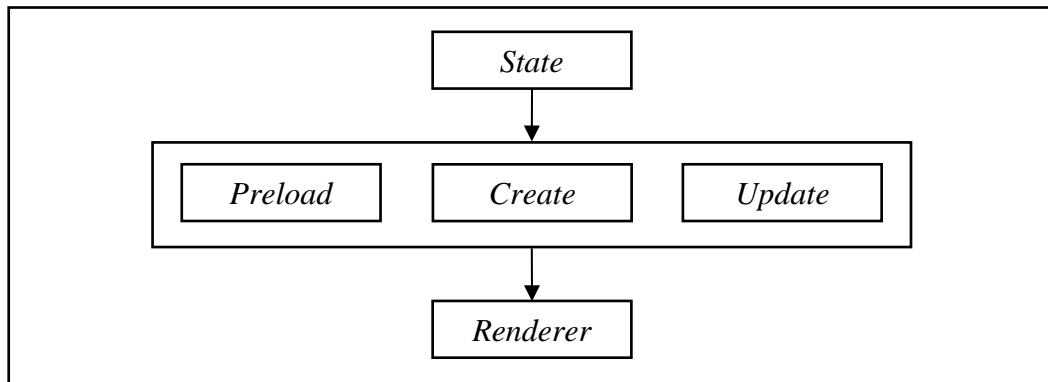
Gambar 3 Tahap merancang *Game* dengan *PixiJS*

4.4 Tahap merancang *Game* dengan *MelonJS*



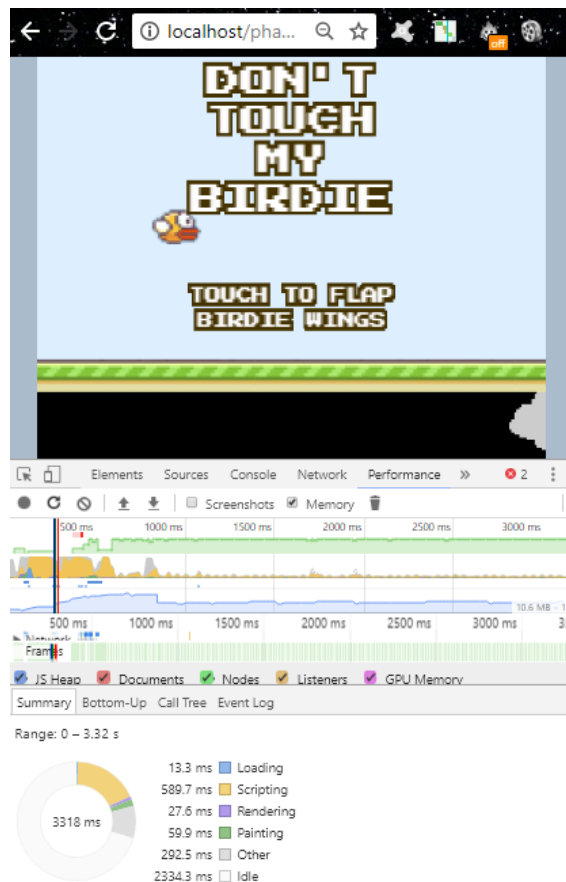
Gambar 4 Tahap merancang *Game* dengan *MelonJS*

4.5 Tahap merancang *Game* dengan *KiwiJS*

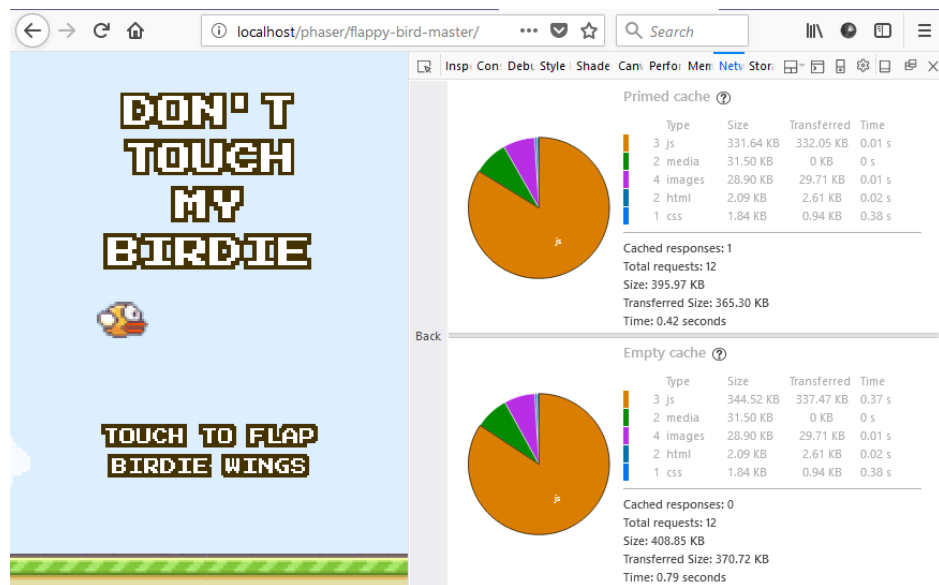


Gambar 5 Tahap merancang *Game* dengan *KiwiJS*

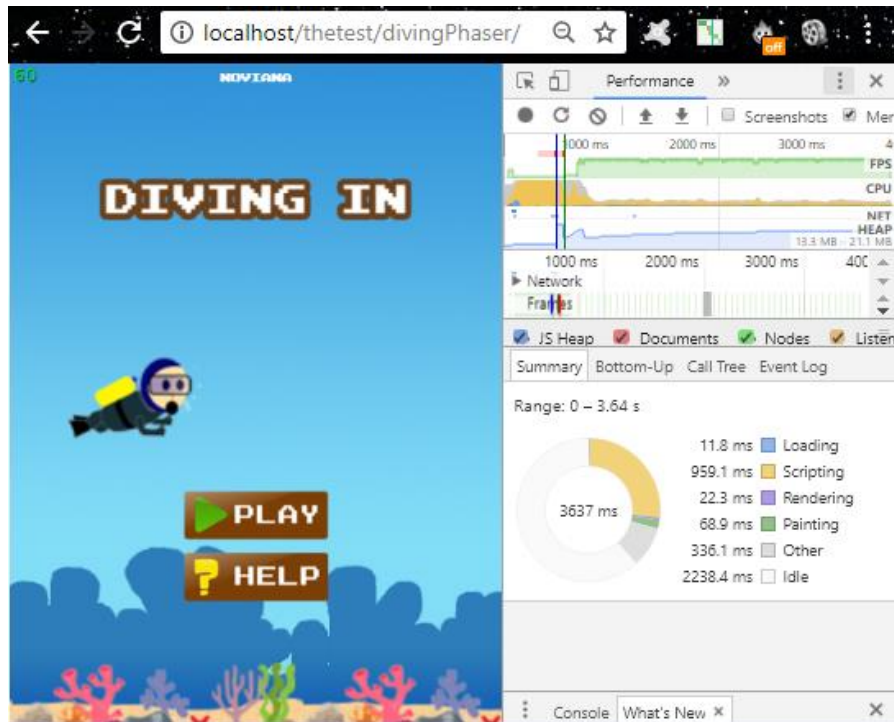
Pengujian game dengan PhaserJS



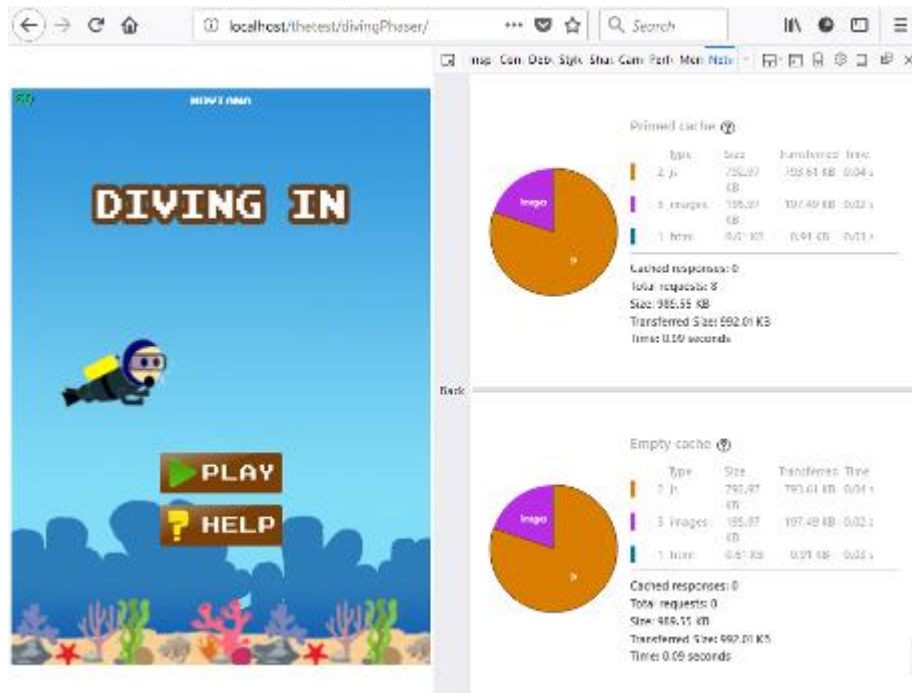
Gambar 6 Performance Flappy Bird (PhaserJS) dengan Chrome Dev Tools



Gambar 7 Performance Flappy Bird (PhaserJS) dengan Mozilla Dev Tools

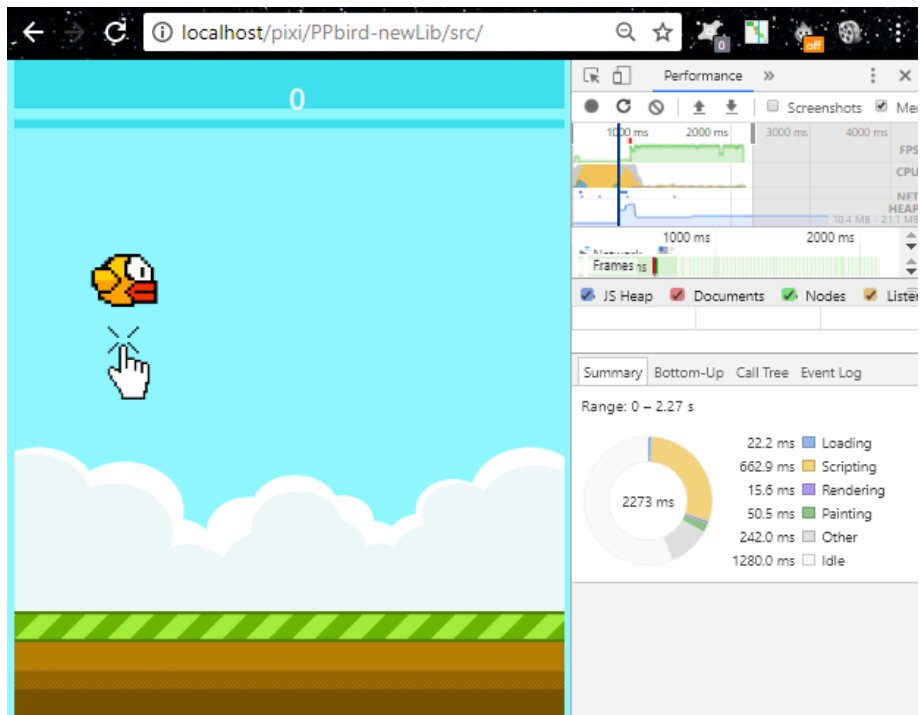


Gambar 8 Performance Diving In (PhaserJS) dengan Chrome Dev Tools

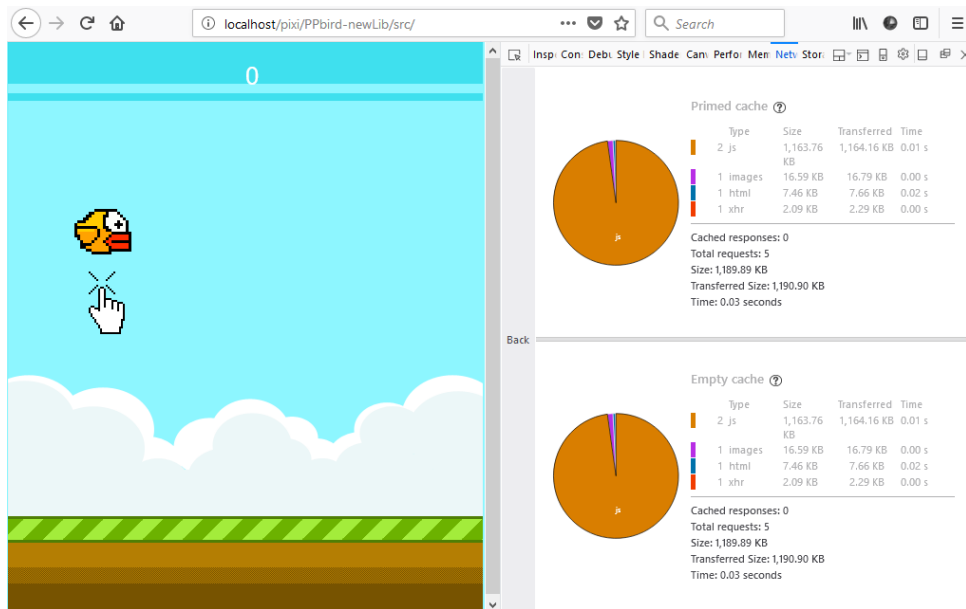


Gambar 9 Performance Diving In (PhaserJS) dengan Mozilla Dev Tools

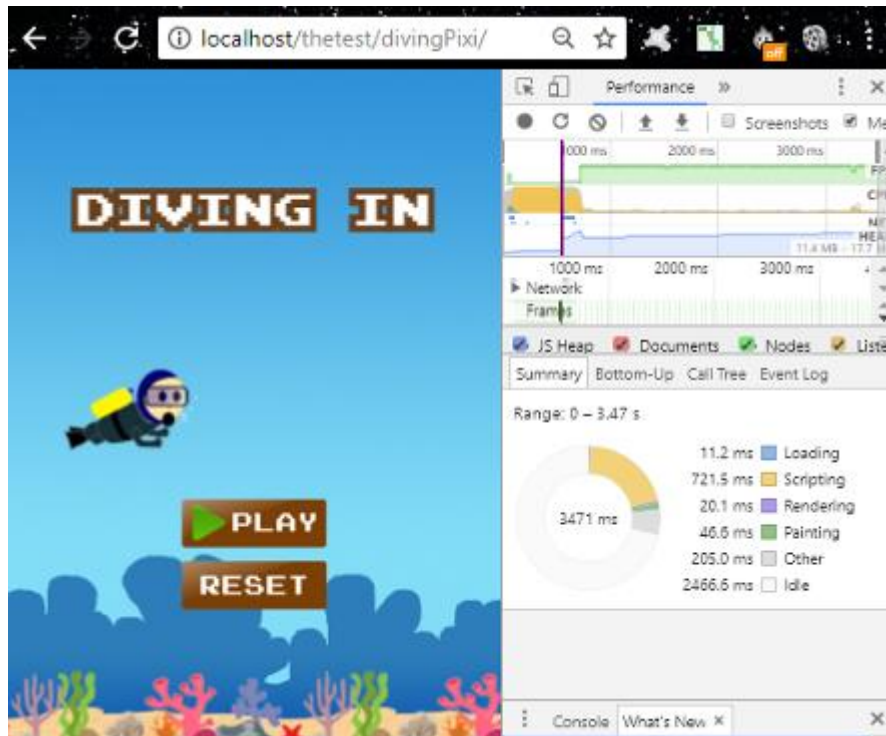
4.6 Pengujian *game* dengan *PixiJS*



Gambar 10 Performance Flappy Bird (PixiJS) dengan Chrome Dev Tools

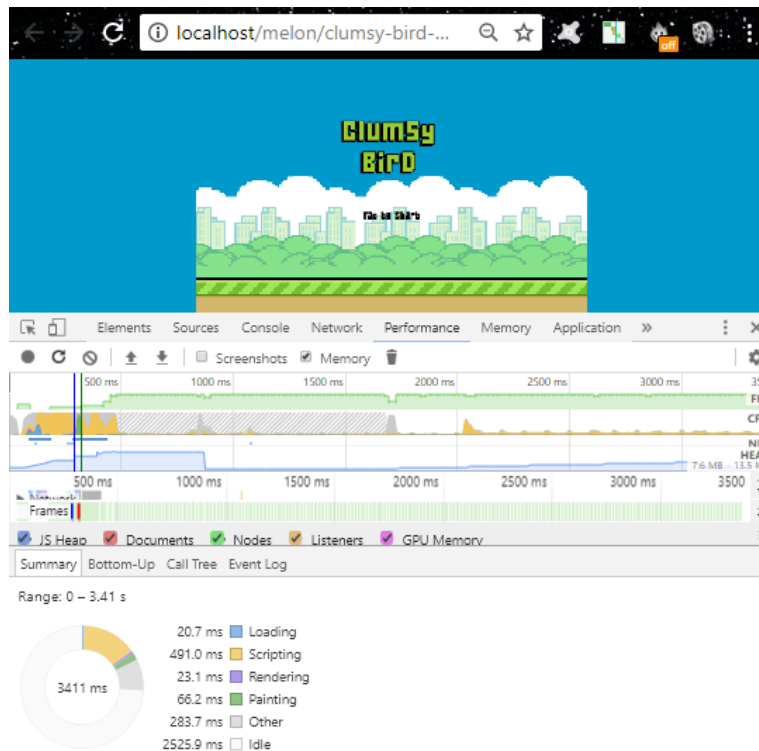


Gambar 11 Performance Flappy Bird (PixiJS) dengan Mozilla Dev Tools



Gambar 12 Performance Diving In (PixiJS) dengan Chrome Dev Tools

4.7 Pengujian game dengan MelonJS



Gambar 13 Performance Flappy Bird (MelonJS) dengan Chrome Dev Tools

4.8 Hasil Perbandingan antar *Game Framework HTML5*

Tabel 1 Hasil Perbandingan

	<i>PhaserJS</i>	<i>PixiJS</i>	<i>MelonJS</i>	<i>KiwiJS</i>
Dokumentasi	Ada	Ada	Ada	Ada
Forum & Diskusi (berdasarkan <i>html5gamedevs.com</i>)	61,136 postingan	13,478 postingan	581 postingan	33 postingan
<i>Performance game Flappy Bird berdasarkan Chrome DevTools</i>				
<i>Loading</i>	13.3 ms	22.2 ms	20.7 ms	14.5 ms
<i>Scripting</i>	589.7 ms	662.9 ms	491 ms	335.6 ms
<i>Rendering</i>	27.6 ms	15.6 ms	23.1 ms	19.9 ms
<i>Painting</i>	46.2 ms	50.5 ms	66.2 ms	84.6 ms
<i>Other</i>	59.9 ms	242 ms	283.7 ms	252.8 ms
<i>Idle</i>	2343.3 ms	1280 ms	2525.9 ms	2762.5 ms
Total	3318 ms	2273 ms	3411 ms	3471 ms
<i>Performance game Diving In berdasarkan Chrome DevTools</i>				
<i>Loading</i>	11.8 ms	11.2 ms	30.1 ms	12.6 ms
<i>Scripting</i>	959.1 ms	721.5 ms	664 ms	584.8 ms
<i>Rendering</i>	22.3 ms	20.1 ms	28.1 ms	27.9 ms
<i>Painting</i>	68.9 ms	46.6 ms	112 ms	146.4 ms

<i>Other</i>	336.1 ms	205 ms	340.5 ms	392.9 ms
<i>Idle</i>	2238.4 ms	2466.6 ms	2474.1 ms	2683.7 ms
Total	3637 ms	3471 ms	3649 ms	3848 ms
<i>Performance game Flappy Bird berdasarkan Firefox DevTools</i>				
<i>Transferred Size</i>	365.30 KB	1190.90 KB	1701.73 KB	720.44 KB
<i>Time</i>	420 ms	30 ms	130 ms	100 ms
<i>Performance game Diving In berdasarkan Firefox DevTools</i>				
<i>Transferred Size</i>	989.55 KB	1403.58 KB	454.43 KB	1,641.07 KB
<i>Time</i>	90 ms	70 ms	120 ms	360 ms

5. KESIMPULAN

Berdasarkan hasil yang didapatkan dalam analisis perbandingan *game framework* pada *HTML5*, penulis menyimpulkan bahwa:

1. *Game framework* yang mempunyai tingkat *performance* terbaik diantara *PhaserJS*, *PixiJS*, *MelonJS* dan *KiwiJS* adalah *gameframework PixiJS* dengan mempunyai waktu tercepat ketika memuat *game* pada *Google Chrome* maupun *Mozilla Firefox*.
2. *Game framework MelonJS* merupakan *game framework* yang sulit untuk digunakan karena pada *game* yang diuji masih banyak komponen yang belum bisa dilengkapi.
3. Pengguna terbanyak dari *game framework* yang dibandingkan, Jika dilihat dari postingan dari forum, *PhaserJS* memiliki jumlah postingan terbanyak yang memungkinkan jumlah pengguna *framework* terbanyak pada saat ini. Selain itu, jumlah postingan atau diskusi yang paling sedikit adalah *game framework KiwiJS*.
4. Dari hasil tingkat *performance*, *KiwiJS* memiliki tingkat *performance* yang paling rendah pada .
5. *PixiJS* juga mempunyai proses *rendering* tercepat dibandingkan dengan *game framework PhaserJS*, *MelonJS* dan *KiwiJS*.
6. Jika menggunakan *Mozilla Firefox* sebagai *platform* dalam menjalankan *game*, *PhaserJS* mempunyai *transferred size* terkecil diantara *game framework* yang dibandingkan lainnya.

Dari kesimpulan di atas, penulis mengajukan beberapa, yaitu:

1. Diharapkan *game framework* bisa dibandingkan dari sisi yang lainnya selain *performance*.
2. Diharapkan *user* yang ingin menggunakan *game framework* ini, setidaknya harus menguasai dasar-dasar algoritma pemrograman, bahasa *HTML5* dan *JavaScript*.
3. Diharapkan *user* mempelajari dokumentasi terlebih dahulu sebelum memulai penggunaan *game framework*.

6. DAFTAR PUSTAKA

Adams, Ernest, *Fundamentals of Game Design Third Edition, New Riders, United States of America*, 2014.

Asmiatun, Siti, Astrid Novita Sari, Belajar Membuat *Game 2D dan 3D* menggunakan Unity, Deepublish, Yogyakarta, 2017.

Faizal, M, Elvianna, Dedy Jauhari, Ricak Agus Setiawan, Moch Rizki Romdoni, Buku Panduan Penulisan Laporan Kerja Praktek Dan Skripsi Sekolah Tinggi Teknologi Indonesia, Tanjungpinang, 2014.

Fulton, Jeff dan Steve Fulton, *The Essential Guide To Flash Games Building Interactive Entertainment with ActionScript*, Friends of ED, United States of America, 2010.

Novak, Jeannie, *Game Development Essentials Third Edition*, Delmar; Cengage Learning, USA, 2012.

Prayitno, Indram, Kupas Tuntas Malware, Elex Media Komputindo, Jakarta, 2006.

Sinarmata, Janner, Rekayasa Perangkat Lunak, CV. Andi OFFSET, Yogyakarta, 2010.

Sutopo, Ariesto Hadi, Multimedia Interaktif dengan Flash, Graha Ilmu, Yogyakarta, 2003.

Tittel, El, Chris Minnick, *Beginning HTML5 and CSS3 For Dummies 1st Edition*, John Wiley & Sons, Hoboken, 2013.

Wicaksono, Yogi, Membangun Bisnis Online dengan Mambo, Elex Media Komputindo, Jakarta, 2008.

Marzuki, Fandri Chandra, Game Berbasis *Adventure* Sebagai Pendukung Pembelajaran Pengenalan Kata Bahasa Inggris Untuk Anak Usia Dini, Universitas Lampung, Bandar Lampung.

WH, Sri Rejeki, Perbandingan Pelaksanaan Program Keluarga Berencana Di Era Desentralisasi Di Kabupaten Lampung Tengah Dan Lampung Barat, Universitas Lampung, Bandar Lampung, 2015.

Adobe Photoshop, [Online] Available:
https://id.wikipedia.org/wiki/Adobe_Photoshop.

Daftar Game Framework, [Online] Available: <https://html5gameengine.com/>.

Definisi *Framework*, [Online] Available:
<https://sheteeguh.wordpress.com/materi-matematika/framework-php/>.

Definisi *Framework*, [Online] Available:
<http://techterms.com/definition/framework>.

Definisi *Framework*, [Online] Available: <http://www.dcc-dp.org/berita498-pengertian-framework-dalam-pemrograman.html>.

Dokumentasi *KiwiJS*, [Online] Available: <http://www.kiwajs.org/api/>.

Dokumentasi *MelonJS*, [Online] Available:
<http://melonjs.github.io/melonJS/docs/>.

Dokumentasi *PhaserJS*, [Online] Available: <http://phaser.io/docs/2.6.2/index>.

Dokumentasi *PixiJS*, [Online] Available:
<http://pixijs.download/release/docs/index.html>.

Fitur *KiwiJS*, [Online] Available: <http://www.kiwajs.org/>.

Fitur *MelonJS*, [Online] Available: <http://melonjs.org/index.html#getting-started>.

Fitur *PhaserJS*, [Online] Available: <https://phaser.io>.

Fitur *PixiJS*, [Online] Available: <http://www.pixijs.com/#features-list>.

Forum *Game Framework*, [Online] Available:
<http://www.html5game devs.com/forum/13-frameworks/>.

Game Clumsy Bird, [Online] Available:
<https://github.com/ellisonleao/clumsy-bird>.

Game Distance Flyer, [Online] Available:
<http://www.kiwajs.org/blueprints/distance-flyer-blueprint/>.

Game Don't Touch My Birdie, [Online] Available:
<https://github.com/uralozden/flappy-bird/>.

Game Copy Flappy Bird, [Online] Available:
<https://github.com/fishenal/PPbird>.

KiwiJS, [Online] Available: <http://www.kiwajs.org/documentation/>.

PhaserJS, [Online] Available: <https://github.com/photonstorm/phaser>.

PixiJS, [Online] Available: <http://pixijs.download/release/docs/index.html>.

MelonJS, [Online] Available: <http://melonjs.org/>.

XAMPP, [Online] Available: <https://id.wikipedia.org/wiki/XAMPP>.