

Pembuatan Game Musical Puzzle “Music Puzzle” Pada Platform Android

Ewaldo Filbert¹⁾ Jeanny Pragantha²⁾ Darius Andana Haris³⁾

^{1) 2) 3)} Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara
Jl. Letjen S. Parman No.1, Jakarta
email: ewaldociady@gmail.com¹⁾, jeannyp@fti.untar.ac.id²⁾, dariush@fti.untar.ac.id³⁾

ABSTRACT

“Music Puzzle” is a musical puzzle game for Android. Player will try to arrange the puzzle piece by listening to the music in each puzzle and correct the order of the music. This game use classical music as the theme of the game. This game is developed by Unity game engine with C# as the programming language. Testing has been done with Black Box Testing, Alpha Testing by lecturer and Beta Testing by survey to 30 respondents. The results show that the “Music Puzzle” is an interesting game to play and encourage respondents to interest in classical music.

Key words

Android Game, Classical Music, Music Puzzle, Puzzle, Unity

1. Pendahuluan

Pada era digital ini, perkembangan teknologi sudah sangat pesat dan penggunaan teknologi sudah tidak asing lagi dan dapat ditemui dalam berbagai bidang. Game merupakan salah satu perkembangan teknologi yang tidak hanya digunakan untuk mencari kesenangan semata tetapi game juga dapat memberikan dampak-dampak positif seperti memperkuat ingatan jangka pendek dan melatih kemampuan memecahkan masalah (*Problem Solving*) [1]. Game itu sendiri adalah permainan yang diprogram pada suatu perangkat yang dapat dijalankan secara *offline* maupun *online*, game memiliki manfaat sebagai sarana hiburan yang dapat menghilangkan penat yang disebabkan oleh rutinitas sehari-hari, selain itu game juga dapat melatih orak untuk memecahkan masalah yang terdapat di dalam game tersebut [2].

Game puzzle merupakan genre game yang menekankan pada pemecahan teka-teki oleh pemainnya seperti permainan logika, pola, urutan, hingga permainan kata. Game Puzzle biasanya tidak terikat dengan peraturan yang banyak sehingga beberapa game puzzle dapat langsung dimainkan oleh pemain yang sebelumnya tidak mengetahui peraturan dari game tersebut [3]. Game Music merupakan genre yang berisi bunyi-bunyian suatu alat musik yang dapat membangkitkan suasana hati pemain saat memainkan gamenya dan game ini juga

dapat menguji atau mengasah pendengaran para pemainnya karena pemain harus peka dengan nada dan tempo dalam *gameplay* [4].

Game yang dirancang memiliki judul *Music Puzzle* yang memiliki genre puzzle dan musik. Judul game ini sudah memiliki arti tersirat yang memberitahu pemain bahwa permainan ini merupakan permainan yang melibatkan musik didalamnya. Musik yang digunakan pada game ini adalah jenis musik klasik. Pertama pemain dapat mendengarkan sebuah musik lalu pemain harus menyusun potongan-potongan musik sehingga tersusun menjadi sebuah musik yang benar. Pemilihan musik klasik pada rancangan ini diharapkan dapat meningkatkan minat orang Indonesia pada musik klasik.

Game dengan rancangan serupa yang sudah dibuat adalah *Rhythm Holic!*. Game ini merupakan game dengan genre *rhythm*. Dalam game ini pemain harus mengikuti irama musik untuk menekan not atau objek yang muncul pada layar. Gambar game dapat dilihat pada Gambar 1 [5].



Gambar 1 Tampilan Game Rhythm Holic

2. Dasar Teori

Dalam perancangan sebuah game ada beberapa hal yang perlu diperhatikan agar dapat menghasilkan game dengan kualitas yang baik. Beberapa hal yang perlu diperhatikan adalah tahap perancangan, genre, game engine yang digunakan, gameplay, sound yang digunakan dalam game.

2.1 Perancangan Game

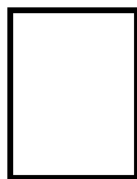
Dalam pembuatan game tentunya dibutuhkan sebuah tahapan perancangan agar game yang dibuat sesuai

dengan yang diinginkan. Tahapan dalam membuat *game* terbagi menjadi[6] :

1. *High Concept*
High Concept pada *game* ini adalah *game single player* dengan tampilan 2D pada platform Android yang berjudul “Music Puzzle”, dibuat menggunakan aplikasi Unity dan C# sebagai bahasa pemrograman. *Game* ini berbahasa Inggris.
2. *Genre*
Game “Music Puzzle” masuk ke dalam gabungan *genre Puzzle* dan *genre Music*. Pada *game* ini, pemain akan diminta untuk menyusun potongan-potongan musik (*puzzle*) yang sudah diacak menjadi susunan musik yang sesuai dengan urutan. sehingga *game* “Music Puzzle” termasuk dalam *genre puzzle*. Pada *game* ini juga menggunakan musik sebagai elemen utama dalam permainan yang akan menguji kepekaan pemain terhadap nada dan tempo, sehingga *game ini* juga termasuk dalam *genre music*. Sebagai tema utama *game* seluruh lagu yang digunakan adalah lagu musik klasik.
3. *Gameplay*
Gameplay menjelaskan tentang bagaimana *game* tersebut dimainkan beserta dengan aturan atau fitur yang ada pada saat dimainkan. Tahap penyusunan *gameplay* terbagi menjadi beberapa tahap sebagai berikut:
 - a. *Control Design*
Control Design pada *game* ini menggunakan *TouchScreen* pada layar *handphone* dengan cara *tap* untuk mendengarkan lagu pada potongan *puzzle* dan *drag and drop* untuk menggeser potongan *puzzle*.
 - b. *Object Design*
 Objek pada *game* ini ada 2 yaitu:
 - I. *Puzzle Piece*
 - II. *Puzzle Box*
 Gambar *Puzzle Piece* dan *Puzzle Box* dapat dilihat pada **Gambar 2 dan 3**.
 - c. *Level Design*
 Pada *game* ini memiliki 3 tingkat kesulitan yaitu *easy*, *medium* dan *hard*. Setiap tingkat kesulitan memiliki 10 lagu berbeda yang dapat dimainkan.
 - d. *Sound Design*
Game ini memiliki 30 lagu yang digunakan sebagai *puzzle* yang harus diselesaikan dan sebuah lagu yang digunakan sebagai lagu background.



Gambar 2 *Puzzle Piece*



Gambar 3 *Puzzle Box*

- e. *Score Design*
 Pada *game* ini menggunakan sistem *score* dengan perolehan bintang, semakin cepat *puzzle* diselesaikan maka semakin banyak bintang yang diperoleh.
4. *Perangkat Keras*
 Perangkat yang dapat digunakan untuk *game* “Music Puzzle” adalah perangkat *mobile* berbasis Android,
5. *Audience*
 Dalam Pembuatan sebuah *game* tentunya ada suatu target golongan yang pembuat inginkan untuk dapat memainkan *game* yang telah dibuat. Target golongan yang dimaksudkan adalah batasan umur agar pemain dapat memainkan *game* ini, golongan dalam *Audience* terbagi atas golongan anak-anak, remaja, dewasa dan semua umur. Target *audience* pada *game* ini dapat dimainkan oleh semua umur.
6. *Rancangan Tampilan*
 Tampilan yang baik diperlukan agar dapat memudahkan pemain berinteraksi dengan *game* yang akan dimainkan. Rancangan tampilan pada *game* ini berisikan gambar – gambar tampilan *user interface* (UI) yang meliputi tampilan menu utama, tampilan pemilihan lagu, tampilan permainan dan tampilan hasil akhir.
7. *Testing*
 Setelah *game* selesai dibuat, maka perlu dilakukan tahap *testing* untuk melihat apakah hasil akhir sudah sesuai dengan konsep dan apakah masih ada hal yang perlu diperbaiki lagi atau ditemukannya error dalam *game*. Tahap testing terbagi menjadi: [7]
 - a. *Alpha Testing*
 - b. *Beta Testing*

2.2 Musik Klasik

Musik klasik biasanya merujuk pada musik klasik Eropa, tapi kadang juga pada musik klasik Persia, India, dan lain-lain. biasanya mengacu pada musik yang berakar dari tradisi kesenian Barat, musik kristiani, dan musik orkestra, mencakup periode dari sekitar abad ke-9 hingga abad ke-21. Musik klasik di Eropa terutama digunakan untuk keperluan lagu di Gereja ataupun lagu untuk pengiringan Raja. Musik klasik memiliki banyak aturan yang harus dipenuhi seperti adanya bentuk-bentuk atau *form* pada musik klasik.[8] Musik klasik memiliki beberapa tipe “standard” *form*. *Form* itu sendiri merupakan sebuah struktur untuk menyusun sebuah musik. Tipe-tipe *form* pada musik klasik yaitu:

1. “ABA” *form*
 ABA *form* merupakan struktur yang paling sederhana dimana *form* ini hanya memiliki 3 *section* yang terdiri dari *section A*, B dan kembali ke A lagi. Pada *section A* biasanya musik terdengar lebih kaku sedangkan *section B* akan terdengar lebih merdu.[9]
2. *Sonata-Allegro form*
 Form ini memiliki 4 *section* yaitu Exposition, Development, Recapitulation dan Coda. Pada

section Exposition akan memainkan *theme* untuk pertama kali. Pada *section* Development, *theme* akan dikembangkan dengan memainkan harmoni pada lagu. Pada *Recapitulation* *theme* awal akan dimainkan kembali. Terakhir pada *section* Coda berisi *ending* yang “exciting”. [10]

3. Rondo Form
Pada *form* Rondo memiliki *Refrain (Main Theme)* yang berganti dengan *Episodes*. *Episode* biasanya kontras dengan *refrain* yang membuat dia menjadi lebih rumit dari *refrain*. Karakteristik dari Rondo itu sendiri akan mendramatisir kembalinya dari *episode* ke *refrain*. Pada *form* Rondo memiliki 5 sampai 7 *section* yang bervariasi, tetapi bentuk paling standarnya adalah *section* ABACA. [11]
4. Theme and Variations
Pada bentuk ini Theme adalah bagian yang akan pertama kali dimainkan dan biasanya memiliki melody yang mudah diingat. Setelah Theme selesai dimainkan, komposer akan memainkan Theme kembali tetapi dengan variasi yang berbeda dan disebut sebagai Variations 1. Setelah Variations 1 selesai dimainkan maka komposer akan kembali memainkan Theme, tetapi dengan variasi yang berbeda dari yang pertama sehingga menghasilkan Variations 2. [12]
5. Sonatina
Pada bentuk ini biasa disebut juga sebagai Sonata kecil. Pada *form* ini memiliki hanya memiliki 3 bagian yaitu *Exposition*, *Development* dan *Recapitulation* yang mirip dengan bentuk sonata. [13]

2.3 Unity

Unity merupakan salah satu *game engine* yang digunakan untuk memudahkan para pengembang *game*. Unity mendukung pembuatan *game* berbasis 2 dimensi. Selain itu, aset gambar objek pendukung yang disediakan pun cukup beragam mulai dari 2 dimensi yang paling sederhana. Selain itu, Unity pun mampu mengambil aset gambar dari perangkat lunak seperti Adobe Photoshop [14]. *Game* “Music Puzzle” dibuat menggunakan Unity.

3. Alur Permainan

Game “Music Puzzle” memiliki 6 modul, yaitu:

1. Modul *Main Menu*
Modul Main menu memiliki 5 tombol yaitu:
 - a. Tombol Play : memulai permainan
 - b. Tombol Help : melihat cara bermain
 - c. Tombol Achievement : menampilkan daftar lagu yang dapat didengarkan setelah menyelesaikan *puzzle*.
 - d. Tombol About : melihat info tentang pembuat beserta dengan sumber lagu yang digunakan.
 - e. Tombol Quit : keluar dari permainan

Tampilan modul main menu dapat dilihat pada **Gambar 4**.

2. Modul *Level Selection*
Pada modul ini digunakan untuk memilih tingkat kesulitan. Tampilan modul Level Selection dapat dilihat pada **Gambar 5**.
3. Modul *Music Selection*
Pada modul ini pemain dapat memilih lagu yang dapat dimainkan. Tampilan modul Music Selection dapat dilihat pada **Gambar 6**.
4. Modul *Gameplay*
Pada Modul ini terdapat:
 - a. Tampilan *Gameplay*
Pada tampilan ini pemain dapat mendengarkan potongan-potongan lagu dan menyusun potongan lagu sesuai dengan urutan. Tampilan *gameplay* dapat dilihat pada **Gambar 7**.
 - b. Tampilan *Pause*
Tampilan *Pause* berada pada tampilan *Gameplay* di bagian kanan atas layar. Tampilan *Pause* dapat dilihat pada **Gambar 8**.
 - c. Tampilan *Stage Clear*
Tampilan *Stage Clear* akan muncul ketika pemain berhasil menyusun *puzzle* dengan benar. Tampilan *Stage Clear* dapat dilihat pada **Gambar 9**.
 - d. Tampilan *Stage Failed*
Tampilan *Stage Failed* akan muncul ketika pemain gagal menyusun *puzzle* dengan benar. Tampilan *Stage Failed* dapat dilihat pada **Gambar 10**.
5. Modul *Achievement*
Pada modul ini akan ditampilkan seluruh lagu yang akan dimainkan dan pemain dapat mendengarkan lagu yang sudah berhasil diselesaikan. Tampilan modul achievement dapat dilihat pada **Gambar 11**.
6. Modul *About*
Modul About terdiri dari:
 - a. Tampilan *About*
About berisikan informasi tentang pembuat dan terdapat sebuah *button credits*. Tampilan *About* dapat dilihat pada **Gambar 12**.
 - b. Tampilan *Credits*
Pada tampilan ini pemain dapat melihat sumber – sumber lagu yang digunakan. Tampilan *Credits* dapat dilihat pada **Gambar 13**.



Gambar 4 Tampilan Main Menu



Gambar 5 Tampilan *Level Selection*



Gambar 10 Tampilan *Stage Failed*



Gambar 6 Tampilan *Music Selection 2*



Gambar 11 Tampilan *Achievement*



Gambar 7 Tampilan *Gameplay*



Gambar 12 Tampilan *About*



Gambar 8 Tampilan *Pause*



Gambar 13 Tampilan *Credits*



Gambar 9 Tampilan *Stage Clear*

4. Hasil Pengujian

Pengujian merupakan tahap yang dilakukan setelah proses pembuatan *game* selesai. Tujuan pengujian untuk memastikan *game* yang dibuat sudah sesuai rancangan dan memastikan *game* yang dibuat dapat berjalan dengan baik.

4.1. Blackbox Testing

Pengujian *Blackbox Testing* dilakukan untuk memeriksa modul-modul yang terdapat pada *game* “Music Puzzle” dengan mencoba seluruh tombol-tombol pada modul apakah sudah berjalan dengan baik atau belum. Berikut adalah modul-modul yang telah diuji:

1. Pengujian Modul *Main Menu*.

Menguji tombol-tombol yang ada pada Main Menu untuk perpindahan scene. Terdapat 5 tombol yang diuji yaitu tombol *play*, *help*, *achievement*, *about* dan *quit*. Tampilan modul *main menu* dapat dilihat pada **Gambar 4**.

2. Pengujian Modul *Level Selection*.
Menguji tombol pemilihan level yaitu tombol *easy*, *medium* dan *hard* yang akan berpindah ke modul *Music Selection*. Tampilan *level selection* dapat dilihat pada **Gambar 5**.
3. Pengujian Modul *Music Selection*.
Pada Modul ini menampilkan daftar lagu yang digunakan. Setiap *level* memiliki 10 lagu yang berbeda. Pengujian dilakukan untuk melihat apakah modul sudah menampilkan daftar lagu yang sesuai dengan *level* yang dipilih dan ketika lagu ditekan dapat pindah ke modul berikutnya. Tampilan *music selection* dapat dilihat pada **Gambar 6**.
4. Pengujian Modul *Gameplay*.
Menguji potongan *puzzle* dapat digeser ke tempat yang sudah disediakan, menampilkan suara ketika potongan *puzzle* ditekan, mencoba tombol *pause* dan *submit* pada *gameplay*. Tampilan *gameplay* dapat dilihat pada **Gambar 7**.
5. Pengujian pada *Pause*
Menghentikan permainan sesaat ketika tombol *pause* ditekan. Terdapat 2 tombol yang dapat ditekan, yaitu tombol *resume* untuk melanjutkan permainan dan tombol *main menu* untuk pindah ke modul main menu. Tampilan *pause* dapat dilihat pada **Gambar 8**.
6. Pengujian pada *Stage Clear*.
Menampilkan *score* dan waktu yang diperoleh oleh pemain ketika berhasil menyusun potongan *puzzle* dengan benar. Pemain dapat menekan tombol *next stage* untuk melanjutkan permainan dengan lagu berikutnya dan tombol main menu untuk kembali ke main menu. Tampilan *stage clear* dapat dilihat pada **Gambar 9**,
7. Pengujian pada *Stage Failed*.
Menampilkan hasil akhir ketika pemain gagal menyusun *puzzle* dengan benar. Terdapat 2 tombol yaitu tombol *retry* untuk memainkan ulang lagu dan tombol main menu untuk kembali ke main menu. Tampilan *stage failed* dapat dilihat pada **Gambar 10**.
8. Pengujian Modul *Achievement*.
Achievement akan menampilkan daftar keseluruhan lagu yang digunakan pada game. Lagu yang sudah berhasil diselesaikan pemain akan menampilkan sebuah tombol disamping judul lagu yang dapat ditekan untuk mendengarkan lagu tersebut. Tampilan modul *achievement* dapat dilihat pada **Gambar 11**.
9. Pengujian Modul *About*.
Menampilkan informasi tentang pembuat dan terdapat sebuah tombol *credits* yang ketika ditekan akan pindah ke modul *credits*. Tampilan *about* dapat dilihat pada **Gambar 12**.
10. Pengujian pada *Credits*

Menampilkan sumber-sumber lagu yang diambil dan ketika ditekan dapat pindah ke laman *youtube*. Tampilan *credits* dapat dilihat pada **Gambar 13**. Hasil pengujian menunjukkan bahwa seluruh modul dan tombol-tombol pada modul telah bekerja dengan baik.

4.2. Alpha Testing

Pengujian *alpha testing* dilakukan secara internal oleh orang yang dapat berperan sebagai perwakilan dari pemain *game*. *Alpha tester* pada *game* ini adalah dosen pembimbing dan Anastasia Aurelia mahasiswi FTI UNTAR yang memiliki gelar *grade 7 ABRSM piano*. Hasil *alpha testing* sebagai berikut:

1. Pada modul *help* jangan menggunakan *scroll down* tapi sebaiknya dibuat menggunakan *button next* untuk lanjut ke instruksi sebelumnya.
 2. Pada modul *music selection* jangan menggunakan angka 1 sampai 10 tapi sebaiknya menampilkan nama komposer dan judul lagu.
 3. Tampilan pada *main menu* terlalu polos dan akan terlihat lebih baik jika menambahkan gambar latar belakang seperti gambar not musik.
 4. Pemotongan lagu sudah baik tetapi masih ada beberapa potongan yang terdengar meleset. Pemotongan dapat dilakukan berdasarkan *phrases*, *sentences* dan *period* pada teori musik.
- Semua kekurangan diatas sudah diperbaiki sehingga *game* bebas dari kekurangan.

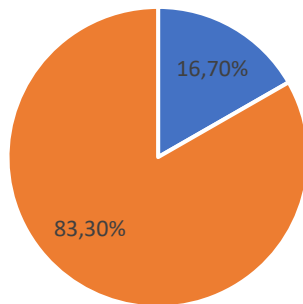
4.3. Beta Testing

Pengujian *beta testing* dilakukan setelah pengujian *alpha testing* selesai dilakukan. *Beta testing* dilakukan secara terbuka kepada siapa saja yang memainkan *game* "Music Puzzle". *Beta testing* dilakukan dengan cara menyebarkan *game* menggunakan *google drive* dan kuisioner menggunakan *google form*, pada tanggal 8 sampai 14 Juni 2020. Beberapa hasil beta testing dapat dilihat pada **Gambar 14** sampai **Gambar 17**.



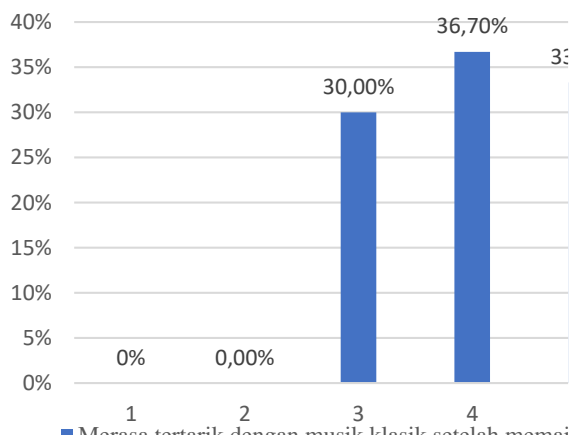
Gambar 14 Pie Chart Genre Game Musik

Pernah memainkan game genre puzzle dan musik



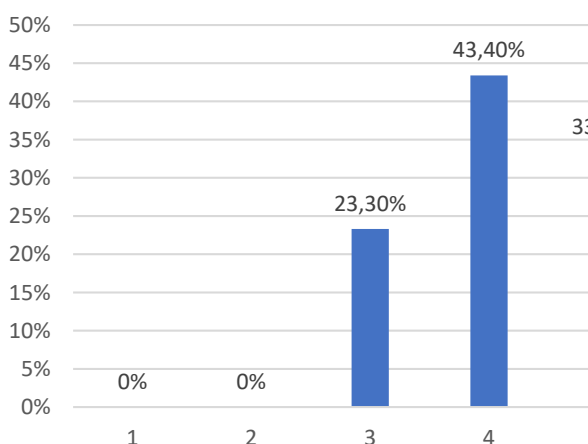
Gambar 15 Pie Chart Genre Game Musik dan Puzzle

Merasa tertarik dengan musik klasik setelah memainkan game



Gambar 16 Diagram Ketertarikan dengan Musik Klasik

Game ini menarik untuk dimainkan lagi



Gambar 17 Diagram Ketertarikan Untuk Memainkan Lagi

4.4. Pembahasan Hasil Pengujian

Pengujian *beta testing* menghasilkan informasi dasar yang dapat menjadi analisis hasil pengujian oleh 30 responden. Berdasarkan dari jawaban responden, terkumpul hasil sebagai berikut:

1. *Game* “Music Puzzle” dapat dimainkan pada *platform* Android dari versi 6 (Marshmallow) hingga Android 10.
2. Sebanyak 93,30% pernah memainkan *game* dengan *genre* musik atau *rhythm* tetapi hanya 5 responden yang pernah memainkan *game* dengan *genre* gabungan musik dan *puzzle*.
3. Responden dapat dikatakan menyukai musik klasik karena secara rata-rata memberi nilai 4 dari skala 5.
4. Dari tingkat kemahiran responden dapat dikatakan bahwa tidak mahir memainkan alat musik karena secara rata-rata memberi nilai 2.
5. Dari tingkat kesulitan rata-rata 10 lagu pada *level easy*, terdapat 8 lagu yang dapat dikatakan mudah dan terdapat 2 lagu yang dapat dikatakan cukup sulit.
6. Dari tingkat kesulitan rata-rata 10 lagu pada *level medium*, seluruh lagu pada *level medium* dapat dikatakan cukup sulit.
7. Dari tingkat kesulitan rata-rata 10 lagu pada *level hard*, seluruh lagu pada *level hard* dapat dikatakan sulit.
8. Help pada *game* “Music Puzzle” membantu pemain untuk memahami cara bermain.
9. Sebanyak 70% responden merasa tertarik dengan musik klasik setelah memainkan *game* “Music Puzzle”. Dari responden tersebut terdapat 81% responden yang tertarik untuk memainkan *game* “Music Puzzle” lagi.

5. Kesimpulan dan Saran

Setelah selesai melakukan pengujian terhadap *game* “Music Puzzle”. Kesimpulan yang dapat ditarik dari hasil pengujian tersebut adalah sebagai berikut:

1. *Game* dengan gabungan *genre music* dan *puzzle* masih jarang ditemukan.
2. Tingkat kesulitan pada *game* “Music Puzzle” sudah sesuai dengan penempatan *level* yang sudah ditentukan.
3. Setelah memainkan *game* “Music Puzzle”, banyak responden jadi lebih tertarik dengan musik klasik sesuai dengan tujuan pembuatan *game* ini yaitu mengenalkan musik klasik.
4. *Game* “Music Puzzle” memiliki tingkat *replayability* yang cukup tinggi karena pemain tertarik untuk memainkan *game* “Music Puzzle” lagi.

Terdapat beberapa saran untuk pengembangan *game* lebih lanjut, yaitu sebagai berikut:

1. Menambahkan fitur *stage unlock* sehingga pemain harus berhasil menyusun lagu pertama terlebih dahulu untuk memainkan lagu kedua.
2. Pada saat proses penyusunan *puzzle* menambahkan fitur untuk menukar posisi 2 *puzzle* sekaligus.

REFERENSI

- [1] Dwi Astarini, Main Puzzle Yuk, Banyak Manfaatnya, <https://merahputih.com/post/read/main-puzzle-yuk-banyak-manfaatnya>
- [2] Helva Silvianita, Pengertian Game Beserta Sejarah, Manfaat, serta Jenis-Jenis Game.Lengkap! <https://www.nesabamedia.com/pengertian-game/>
- [3] School Pouring Rights, Puzzle, Permainan Sederhana Namun Memiliki Banyak Manfaat, <http://www.schoolpouringrights.com/unik/puzzle-permainan-sederhana-namun-memiliki-banyak-manfaat/>
- [4] Andi Taru, Apa itu Genre Game?, <https://www.gamelab.id/news/15-apa-itu-genre-game>
- [5] Jastin Ng, Jeanny Pragantha, Darius Andana Haris, 2020, Pembuatan Game Rhythm “Rhythm Holic!” Pada Platform Android, Jurnal Ilmu Komputer dan Sistem Informasi, Jakarta: Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Tarumanagara.
- [6] Bob Bates, Game Design Second Edition, (Stamford: Cengage Learning PTR, 2004), h. 204.
- [7] Luke Freiler, Alpha Testing vs Beta Testing, <http://www.centercode.com/blog/2011/01/alpha-vs-beta-testing/>,
- [8] Dardus Gandute, Ciri-Ciri Musik Klasik Beserta Karakteristik dan Penjelarasannya, <https://www.seluncur.id/ciri-ciri-musik-klasik/>, 24 Maret 2020
- [9] Lumen, Ternary Form, <https://courses.lumenlearning.com/atd-epcc-musicappreciation/chapter/ternary-form/>,
- [10] Lumen, The Sonata Allegro Form, https://courses.lumenlearning.com/musicappreciation_with_theory/chapter/the-sonata-allegro-form/,
- [11] OpenMusicTheory, Rondo, <http://openmusictheory.com/rondo.html>
- [12] Ben, Theme and Variations, <https://www.musictheoryacademy.com/understanding-music/theme-and-variations/>,
- [13] The Editors of Encyclopaedia Britannica, <https://www.britannica.com/art/sonatina-music>, 28 Juni 2020
- [14] Adhitya Wibawa Putra, Unity – Game Engine Tangguh Untuk Berbagai Platform, <https://teknojurnal.com/unity/>,

Ewaldo Filbert, mahasiswa S1, program studi Teknik Informatika, Fakultas Teknologi Informasi Universitas Tarumanagara.

Ir. Jeanny Pragantha, M.Eng, memperoleh Ir dari institute Teknologi Bandung. Kemudian memperoleh gelar M. Eng. Dari Asian Institut of Technology, Bangkok. Saat ini sebagai dosen program studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara.

Darius Andana Haris, M.TI, memperoleh gelar S.Kom dari Universitas Tarumanagara pada 2009, melanjutkan S2 di Universitas Bina Nusantara dan memperoleh helar M.TI. Saat ini sebagai dosen program studi Teknik Informatika, Fakultas Teknologi Informasi Universitas Tarumanagara.