

RANCANG BANGUN APLIKASI PERMAINAN ADU CEPAT MEMBACA MENGGUNAKAN PERMAINAN *MULTIPLAYER WORD SEARCH PUZZLE*

Muhammad Sholeh, Erfanti Fakhia, Gilangsurya Adi P.

Teknik Informatika, Fakultas Teknologi Industri
Institut Sains & Teknologi AKPRIND Yogyakarta
Jl. Kalisahak 28 Komplek Balapan Yogyakarta
Email : muhash@akprind.ac.id

Abstract

Various mcam mobile-based applications today are easily downloaded and played via mobile devices. One application that can be used is a game application, both games for entertainment and games in which there is an element of education. With a megemas game that there is an element of education is expected, players, especially children in addition to getting entertainment also get education, Indirectly children learn and have to improve the ability that actually there element of education.

This paper discusses the application of fast reading games by combining fast reading techniques and word search. This application is built using the Java Programming Language and the server is built with the Javascript Programming Language using the Node.js framework.

Algorithm to solve game search word in this research using backtracking algorithm. To play this game requires two smartphone devices connected to each other, for that required media connection between devices ie the internet. The design and manufacture of this game aims to train the ability to read quickly as well as entertainment to fill the spare time.

Keywords: *games, quick reading, word search games, backtracking, android*

Pendahuluan

Latar Belakang

Perangkat *mobile* pada saat ini telah berkembang menjadi *smartphone* (telepon pintar) yang tak lepas dari dukungan sistem operasi seperti iOS dan Android. Tidak hanya sekedar digunakan sebagai alat komunikasi, teknologi *smartphone* dapat memudahkan pengguna dalam kehidupan sehari-hari termasuk digunakan sebagai media untuk pendidikan. Hal ini dapat terjadi karena teknologi *smartphone* memiliki banyak fasilitas, antara lain pengaksesan internet, *e-mail*, GPS, pemutar music maupun video, kamera, permainan (*game*), dan lain sebagainya yang dapat digunakan di mana saja dengan mudah.

Dari banyaknya fasilitas yang dapat digunakan pada *smartphone*, permainan merupakan salah satu fasilitas penghibur alternatif bagi pengguna *smartphone*. *Game* saat ini sudah dapat dikatakan sebagai permainan *universal*. Mulai dari anak-anak, remaja sampai dengan orang dewasa pun sudah tidak merasa asing lagi dengan *game*. Permainan atau *game* pada *smartphone* juga sangat banyak dan beraneka ragam,

banyaknya *genre* permainan *smartphone* seperti petualangan, *arcade*, adu kecepatan, strategi, dan lain sebagainya dari yang paling mudah dimainkan hingga yang paling sulit.

Permasalahan

Alternatif dalam metode pembelajaran diantaranya adalah mengemas dengan permainan sehingga selama melakukan permainan pengguna juga melakukan proses pendidikan. Salah satu upaya yang dilakukan adalah pengembangan game aplikasi untuk melihat sejauh mana ketangkasan dalam membaca cepat sebuah kata. Permasalahan dalam rancang bangun aplikasi ini adalah bagaimana merancang dan membangun aplikasi untuk membantu mengasah dan mengadu kemampuan membaca cepat yang dikemas dalam sebuah permainan *word search puzzle* (cari kata) dengan menerapkan algoritma runut balik (*backtracking*).

Tujuan Penulisan

Rancang bangun aplikasi ini bertujuan untuk memberikan alternatif dalam belajar membaca cepat yang dikemas dalam permainan, permainan ini dapat menguji kemampuan membaca dan mencari informasi yang terkandung dalam teks bacaan dengan cepat serta mengimplementasikan teknik desain permainan *word search puzzle* (cari kata) menggunakan algoritma runut balik (*backtracking*) dalam pembuatan *game smartphone*.

Landasan Teori Game (Permainan)

Game (permainan) adalah suatu sistem atau program dengan satu atau lebih pemain mengambil keputusan melalui kendali pada objek di dalam permainan untuk tujuan tertentu (Jasson, 2009).

Seiring dengan perkembangannya, *game* menghasilkan beberapa klasifikasi atau *genre* yang didasarkan pada interaksinya. Visualisasi juga menjadi ukuran klasifikasi *genre*. Namun untuk beberapa kasus pengembang *game* membuat kompilasi antar berbagai *genre*. Tentu saja variasi format *game* lebih banyak (Sibero, 2009). Permainan tersebut ada yang dimainkan secara sendirian (*single player*) dan ada yang dapat dimainkan lebih dari satu orang (*multiplayer*).

Membaca Cepat

Membaca cepat merupakan jenis membaca yang mengutamakan kecepatan, dengan tidak meninggalkan aspek bacaannya. Biasanya kecepatan itu dikaitkan dengan tujuan membaca, keperluan dan bahan bacaan. Dalam membaca cepat tidak hanya membaca dengan cepat saja, akan tetapi juga harus mampu memahami isi bacaan yang dibaca secara cepat pula (Nurhadi, 2005).

Word Search Puzzle (Cari Kata)

Word Search Puzzle (Cari Kata) merupakan permainan *puzzle* pencarian kata-kata tersembunyi yang disusun dalam bentuk *array* dua dimensi (*matriks*). Objektif dari permainan ini adalah untuk mencari kata yang tersembunyi di *matriks* permainan tersebut. Kata-kata tersebut dapat disusun secara horizontal, vertikal, dan diagonal dan dapat ditulis dalam posisi terbalik maupun tidak. Gambar 1 merupakan matrik susunan kata

M	A	S	C	I	E	N	T	I	S	T	O	E	K
O	M	D	B	K	O	O	B	C	I	M	O	C	D
G	Y	N	R	K	O	N	P	S	P	R	H	A	Y
O	W	L	E	O	N	A	R	D	E	S	N	P	T
S	C	N	A	D	S	E	T	L	E	H	B	S	I
E	A	D	V	A	I	E	N	J	S	H	I	P	S
D	D	O	D	T	Y	P	A	O	O	O	G	H	R
C	Y	E	S	K	N	R	E	A	L	W	B	Y	E
E	N	S	E	C	C	E	O	N	N	A	A	S	V
A	P	E	N	N	Y	H	N	E	G	R	N	I	I
Z	T	I	W	O	L	O	W	T	H	D	G	C	N
E	C	N	E	I	C	S	S	N	C	T	N	S	U
A	G	R	P	P	Y	S	H	E	L	D	O	N	P
E	E	N	T	E	T	M	E	W	N	T	I	S	T

Gambar 1 Contoh permainan *word search puzzle*

Permainan ini membutuhkan tingkat ketelitian yang tinggi untuk menemukan kata yang dicari dalam sekumpulan huruf acak pada tabel. Semakin besar ukuran tabel dan semakin pendek ukuran kata yang dicari maka tingkat kesulitannya bertambah.

Algoritma Runut Balik (*Backtracking*)

Algoritma runut balik (*backtracking*) merupakan algoritma yang digunakan untuk mencari solusi persoalan secara lebih mangkus daripada menggunakan algoritma *brute force*. Algoritma ini cukup mangkus untuk digunakan dalam beberapa penyelesaian masalah dan juga untuk memberikan kecerdasan buatan dalam *game*. Beberapa *game* populer semisal Sudoku, Labirin, Catur juga bisa diimplementasikan dengan menggunakan algoritma runut balik (Munir R. , 2009).

Algoritma ini akan mencari solusi berdasarkan ruang solusi yang ada secara sistematis namun tidak semua ruang solusi akan diperiksa, hanya pencarian yang mengarah kepada solusi yang akan diproses.

Algoritma runut balik berbasis pada DFS (*Depth First Search*) sehingga aturan pencariannya akan mengikut kepada aturan pencarian DFS yaitu dengan mencari solusi dari akar ke daun (dalam pohon ruang solusi) dengan pencarian mendalam. Simpul-simpul yang sudah dilahirkan (diperiksa) dinamakan simpul hidup (*live node*). Simpul hidup yang sedang diperluas dinamakan simpul-E atau *Expand Node*. (Munir R. , 2009)

Algoritma *backtracking* mempunyai prinsip dasar yang sama seperti *brute-force* yaitu mencoba segala kemungkinan solusi. Perbedaan utamanya adalah pada ide dasarnya, semua solusi dibuat dalam bentuk pohon solusi (pohon ini tentunya berbentuk abstrak) dan algoritma akan menelusuri pohon tersebut secara *DFS (depth field search)* sampai ditemukan solusi yang layak (Munir R. , 2009).

Studi Pustaka

Pengembangan aplikasi permainan berbasis android sudah cukup banyak yang mengembangkan, beberapa hasil studi pustaka yang terkait dengan permainan dengan android adalah (Putra, Nugroho, & Puspitarini, 2016) dalam tulisannya yang dimuat di jurnal JIMP mengupas tentang penggunaan ponsel pintar (smartphone), Ipad, Tablet PC, dan lain sejenisnya dalam kehidupan sehari-hari kian marak. Mulai merk terkenal

sampai biasa saja, dan yang canggih sampai yang sederhana kian santer digunakan. Berbagai alasan menjadi magnet bagi semua kalangan dalam penggunaannya. Ringan, mudah dibawa, praktis menjadi alasan utama pemakaiannya. Selain itu, Terdapat banyak fitur game yang tidak hanya sebagai hiburan bermain, namun sudah banyak game untuk mengasah daya pikir dan logika yang dapat memperkenalkan materi agar lebih menarik untuk diterima dan dipahami terutama oleh anak yang masih dalam usia dini. Anak usia dini antara umur 3–6 tahun berada dalam masa Golden Periode (Periode Keemasan) perkembangan otak mereka. Dalam usia ini, mereka berada pada masa pertumbuhan dan perkembangan yang paling pesat baik fisik maupun mental. Oleh karena itu, penulis membuat game edukasi yang berbasis sistem operasi Android dengan harapan anak-anak dapat langsung menggunakan aplikasi permainan tersebut dan memperoleh pengetahuan lebih banyak serta merubah pola belajar agar tidak jenuh dan bosan. Game edukasi ini merupakan aplikasi pembelajaran untuk anak usia dini dimulai dari usia 3 sampai 6 tahun yang berisi tentang materi pelajaran mengenal binatang, mewarnai, corat-coret, menyanyi serta alfabet. Metode penelitian dan pengembangan aplikasi edukasi ini adalah metode Waterfall yang terdiri dari lima tahapan yaitu Requirement, Design, Implementation, Verification, dan Maintenance. Dengan menerapkan hasil dari game edukasi ini, diharapkan dapat membantu anak-anak dalam belajar dan dapat meningkatkan pola pikir kreatif serta menambah pengetahuan lebih maju.

Perkembangan teknologi terutama *smartphone*, *tablet*, *ipad* dan sebagainya menjadi sangat populer saat ini. Mulai dari kebutuhan pengguna yang hampir lengkap dengan sebuah benda kecil dengan berjuta manfaat. *Smartphone*, *tablet* dan *ipad* telah banyak digunakan oleh semua kalangan pengguna. Pengguna *smartphone*, *tablet* dan *ipad* tidak hanya orang-orang dewasa melainkan anak-anak sudah diberikan fasilitas tersendiri yang dapat digunakannya untuk bermain, namun tidak hanya bermain sudah banyak aplikasi-aplikasi permainan yang mengandung unsur edukasi atau pembelajaran. *Edu Game* merupakan aplikasi belajar dengan menerapkan permainan didalamnya. Dimana, anak-anak tidak hanya dapat bermain tetapi mendapatkan pengetahuan dari permainan tersebut. Strategi belajar yang tidak membuat anak bosan dengan pelajaran yang didapatkan dari sekolah dan buku pelajaran namun, kini anak-anak dapat mengembangkan pola pikir lebih kreatif dan melihat benda atau gambar dalam bentuk *digital*. Oleh karena itu, penulis membuat aplikasi edukasi “Ceria” dengan harapan anak-anak dapat langsung menggunakan aplikasi permainan tersebut dan memperoleh pengetahuan lebih banyak serta merubah pola belajar agar tidak jenuh dan bosan. Aplikasi edukasi “Ceria” merupakan aplikasi pembelajaran untuk anak usia dini dimulai dari usia 2 sampai 5 tahun. Aplikasi ini berisi materi-materi pelajaran mengenal huruf, angka, kata dan berhitung berupa gambar dan suara. Tidak hanya itu, aplikasi ini disajikan seperti layaknya sebuah *game* yang dibuat agar anak-anak cenderung tidak bosan dalam belajar. Aplikasi ini dibangun menggunakan sistem operasi android. Metode penelitian yang digunakan penulis dalam penyusunan skripsi ini adalah metode RUP (*Rational Unified Process*). Dimana hasil dari perancangan aplikasi ini yaitu sistem aplikasi edukasi bagi anak-anak yang diharapkan dapat membantu anak dalam belajar dan dapat meningkatkan pola pikir kreatif serta menambah pengetahuan lebih dari aplikasi tersebut. (P, Apriani, & v, 2014)

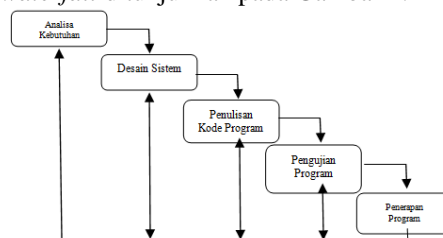
(Firdaus & Nugroho, 2015) Pada era modern seperti sekarang ini, perkembangan mengenai teknologi sangat pesat. Apalagi teknologi dalam bidang komunikasi seperti gadget atau handphone yang berbasis Sistem Operasi Android. Dengan perkembangan seperti itu sangat membantu manusia dalam mendapatkan informasi maupun untuk kehidupan sehari-hari. Tidak hanya untuk kalangan orang dewasa, perkembangan teknologi juga dapat dirasakan untuk kalangan anak kecil. Seperti pembelajaran untuk anak yang menggunakan permainan.

Game edukasi adalah pembelajaran anak yang menggunakan metode permainan yang menarik dan menyenangkan. Banyak sekali aplikasi sekarang ini yang sangat membantu untuk membuat game edukasi tersebut, salah satunya adalah Construct 2. Dengan menggunakan Construct 2, yaitu aplikasi untuk membuat desain dan animasi game “Asah Otak Anak” berbasis android menjadi lebih menarik dan menyenangkan.

Metode Penelitian

Metode pengumpulan data yang dilakukan yang bertujuan untuk memperoleh data yang sesuai dan diharapkan membantu dalam pembuatan game agar tepat sasaran. Metode pengumpulan data yang digunakan adalah Metode Observasi, merupakan metode pengumpulan data yang dilakukan dengan cara melakukan pengamatan secara langsung pada objek yang diteliti, kemudian dilakukan pencatatan hal-hal yang berhubungan dengan objek yang diteliti. Studi pustaka, merupakan metode pengumpulan data dengan cara membaca buku-buku literatur yang ada hubungannya dengan permasalahan yang dijadikan objek penelitian.

Metode pengembangan Perangkat Lunak Langkah dan diagram alir penelitian merupakan penjelasan dari suatu informasi yang terdiri dari suatu sistem informasi secara utuh ke dalam komponennya dengan maksud mengidentifikasi dan mengevaluasi permasalahan-permasalahan, hambatan dan kebutuhan yang diperlukan, sehingga dapat diusulkan perbaikan. Langkah-langkah yang dilakukan dalam penelitian ini menggunakan metode *waterfall* ditunjukkan pada Gambar 2.



Gambar 2 Diagram alir penelitian

Alat Penelitian

Perangkat keras (*Hardware*)

Perangkat keras yang digunakan dalam merancang dan membangun *game* ada membaca cepat ini adalah *notebook* dengan spesifikasi minimal sebagai berikut untuk *Processor* menggunakan Intel® Core(TM) i3-6006U CPU@ 2.00 GHz 4 CPU, *Memory* sebesar 4.00 GB RAM, *Hard disk* 500 GB, Layar 14 inch WXGA LED, resolusi maksimal 1366 x 768, *Graphic card* NVIDIA GeForce 920MX.

Perangkat lunak (*Software*)

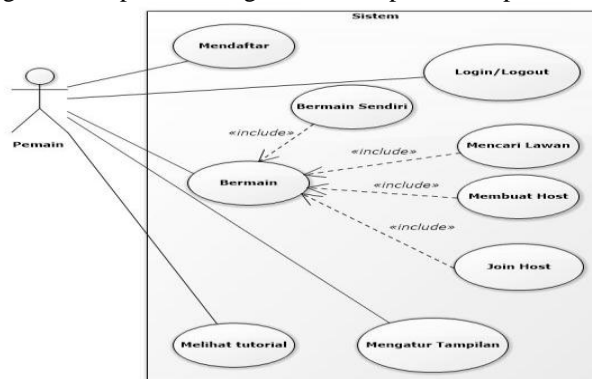
Spesifikasi minimal perangkat lunak yang digunakan untuk merancang dan membangun *game* ada membaca cepat ini adalah Sistem Operasi Windows 10 Ultimate

64 bit, Android Studio 3.0.1 digunakan untuk pembuatan aplikasi Android, WebStorm 2017.3.2 digunakan untuk pembuatan aplikasi *server*, Perangkat Uji yang digunakan untuk memainkan permainan adu membaca cepat secara *multiplayer* yaitu Android Nougat 7.1.2 API 25 dan Android Jelly Bean 4.2.2 API 17

Perancangan UML

Use Case Diagram Aplikasi

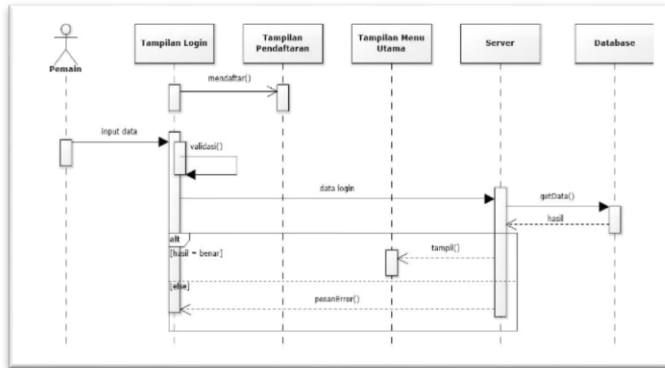
Pada perancangan *use case diagram* dalam permainan ini pemain diharuskan untuk mendaftar untuk mendapatkan akun dan *username* untuk membedakan antara pemain lain. Pemain yang sudah memiliki akun dapat melakukan proses *login* dan *logout*. Pemain dapat melihat petunjuk/*tutorial* permainan dan mengatur tampilan aplikasi. Untuk bermain, pemain dapat memilih tipe permainan yang tersedia, antara lain: bermain sendiri, mencari lawan bermain, membuat *hot*, atau *join host*. Untuk pilihan “bermain sendiri”, pemain akan diberi jangka waktu tertentu untuk menyelesaikan permainan. Untuk pilihan “mencari lawan”, sistem permainan akan mencarikan lawan yang sama-sama memilih pilihan “mencari lawan” untuk dijadikan lawan. Untuk pilihan “membuat *host*”, pemain akan diberi kode unik yang dapat digunakan oleh pemain yang memilih pilihan “*join host*” untuk bermain melawan *host* tersebut. Hubungan antara pemain dengan sistem dapat dilihat pada Gambar 3



Gambar 3 Use case diagram pemain dengan sistem

Sequence Diagram Proses Login

Pada Gambar 4 dapat dilihat alur pemain atau *client* untuk melakukan proses *login*. Proses diawali dengan memasukkan data yang kemudian sistem akan me validasi persyaratan data tersebut. Apabila persyaratan kurang, akan ditampilkan pesan *error* dan apabila persyaratan sudah benar, akan menuju proses berikutnya. Proses berikutnya yaitu pengiriman data ke *server* yang kemudian akan mengecek ke *database*. *Database* kemudian akan memberikan hasil: Apabila hasil benar berarti proses *login* sukses dan pemain akan dialihkan menuju menu utama; jika hasil salah, akan ditampilkan pesan *error* pada tampilan *login*. Dari tampilan *login* ini, pemain juga dapat beralih ke tampilan pendaftaran.

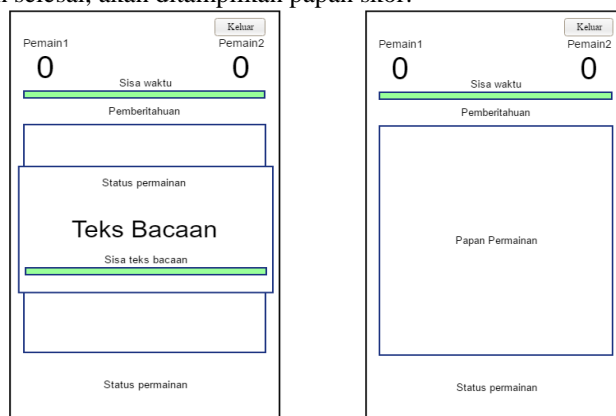


Gambar 4 *Sequence diagram* proses login

Perancangan Tampilan

Perancangan tampilan merupakan rancangan tatap muka antara pemain/*user* dengan sistem agar memudahkan *user* menggunakan aplikasi. Perancangan *interface* pada penelitian ini.

Perancangan tampilan permainan terdiri dari bagian skor, papan permainan, dan tampilan membaca. Ketika sesi membaca dimulai akan muncul *pop-up* tampilan membaca yang akan menampilkan teks bacaan seperti yang terlihat pada Gambar 4. Setelah sesi membaca selesai, tampilan *pop-up* tadi akan tertutup. Pada tampilan permainan yang ditunjukkan oleh Gambar 5, terdapat papan skor yang berisi nama pemain dan juga skor yang diperoleh. Di bawah papan skor terdapat *progress bar* yang menunjukkan sisa waktu untuk mencari jawaban pada saat sesi menjawab. Bagian papan permainan berfungsi untuk menampilkan permainan cari kata/ *word search puzzle*. Tombol “keluar” digunakan apabila pemain ingin meninggalkan permainan. Setelah permainan selesai, akan ditampilkan papan skor.



Gambar 5 Rancangan tampilan permainan sesi membaca dan tampilan permainan sesi menjawab

Perancangan Basis Data

Rancangan *database* merupakan tahap penentuan tabel dan beserta atribut yang digunakan. Rancangan *database* bertujuan untuk membentuk struktur tabel. Tabel yang akan diimplementasikan dalam penelitian ini antara lain:

1. Nama Tabel : users

Tabel 1 Tabel users

No.	Nama Field	Tipe
1	id	Int32
2	username	String
3	password	String
4	modified	Date

Tabel users berfungsi untuk menyimpan data akun pemain.

2. Nama Tabel : game_material

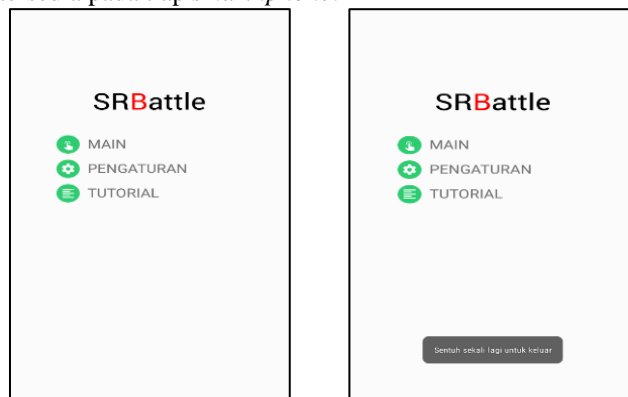
Tabel 2 Tabel game_material

No.	Nama Field	Tipe
1	id	ObjectId
2	teks_bacaan	String
3	pertanyaan1	String
4	jawaban1	String
5	pertanyaan2	String
6	jawaban2	String
7	pertanyaan3	String
8	jawaban3	String

Tabel game_material berfungsi untuk menyimpan bahan-bahan yang digunakan dalam permainan. Bahan-bahan tersebut antara lain: teks bacaan, soal pertanyaan, dan jawaban.

Hasil Dan Pembahasan

Tampilan awal dari aplikasi ini menampilkan menu utama untuk proses verifikasi *login*. Pada tampilan ini terdapat tiga buah tombol yaitu: tombol “MAIN”, tombol “PENGATURAN”, dan tombol “TUTORIAL” seperti yang terlihat pada Gambar 6. Tombol ini berfungsi untuk mengalihkan menuju tampilan yang dipilih. Dari tampilan ini pengguna juga dapat menutup aplikasi dengan cara menyentuh dua kali tombol kembali yang tersedia pada tiap *smarthphone*.

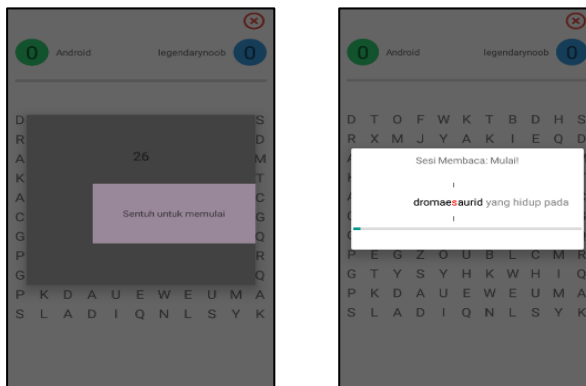


Gambar 6 Tampilan menu utama dan tampilan pemberitahuan keluar aplikasi

Permainan dan Cara Bermain Pertarungan Membaca Cepat

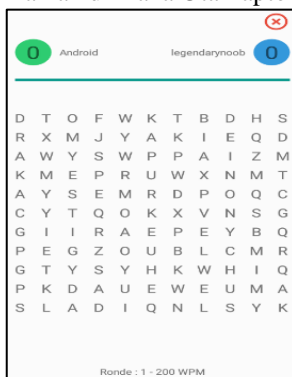
Setelah data materi permainan diterima, aplikasi akan memprosesnya data tersebut ke dalam permainan. Sebelum memulai permainan, terdapat tampilan dialog *pop-up* konfirmasi bermain seperti yang terlihat pada Gambar 6. Sentuh tombol “Sentuh untuk memulai” untuk memulai permainan. Untuk mode bermain melawan pemain lain, permainan akan dimulai ketika kedua pemain telah menyentuh tombol tersebut. Apabila selama tiga puluh detik salah satu pemain atau tidak ada pemain yang menyentuh tombol tersebut maka permainan dibatalkan dan pengguna akan dialihkan ke tampilan *lobby*.

Proses berikutnya yaitu sesi membaca. Pada sesi ini ditampilkan dialog *pop-up* yang berupa potongan teks bacaan seperti yang terlihat pada Gambar 7. Teks bacaan tersebut akan bergerak ke arah kiri ketika permainan dimulai dan berhenti ketika mencapai kata terakhir. *Progress* dari proses membaca tersebut juga dapat dilihat pada *progress bar* yang terletak di bawah teks bacaan.



Gambar 7 Tampilan konfirmasi bermain dan Tampilan sesi membaca

Untuk memperjelas proses permainan, digunakan salah satu materi *game* yang dipakai pada permainan ini. Setelah sesi membaca selesai, dilanjutkan dengan sesi pertanyaan pertama. Pada sesi ini juga akan ditampilkan sebuah teks pertanyaan yang juga bergerak seperti pada sesi membaca. Teks pertanyaan pertama yang digunakan pada contoh kasus ini adalah: "Zaman di mana Utahraptor hidup".



Gambar 8 Tampilan papan permainan

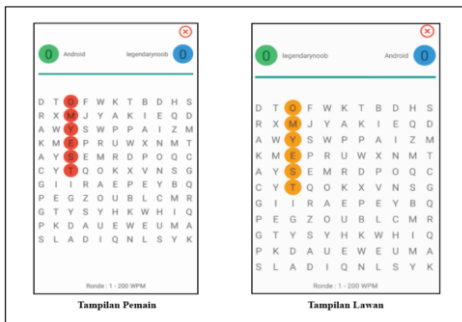
Setelah sesi pertanyaan selesai, dialog *pop-up* teks pertanyaan tersebut akan tertutup dan akan terlihat tampilan papan permainan seperti yang terlihat pada Gambar 8. Pada tampilan ini terdapat papan skor yang berisi nama dan skor yang diperoleh masing-masing pemain, *progress bar* yang menunjukkan sisa waktu kesempatan menjawab yang terletak di bawah papan skor, dan papan permainan yang digunakan untuk mencari jawaban.

Pada permainan ini ditampilkan tabel berukuran 11x11 yang berisi huruf acak dan di dalam salah satu bagian huruf acak tersebut tersusun jawaban dari pertanyaan yang ditanyakan pada sesi pertanyaan sebelumnya. Pada penelitian ini, digunakan *grid view* yang telah tersedia pada pemrograman Android untuk menyusun data papan permainan dari server menjadi tampilan papan permainan pada aplikasi Android.

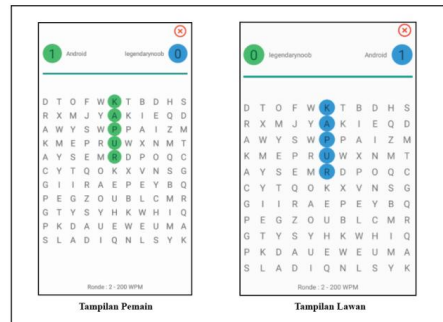
Pada sesi menjawab, pemain harus mencari jawaban pada papan permainan dengan cara menyentuh huruf pertama dari jawaban dan geser ke arah huruf terakhir dalam posisi horizontal, vertikal, atau diagonal sebelum mencapai batas waktu selama tiga puluh detik. Susunan huruf yang terbentuk akan berubah warna menjadi berwarna merah untuk pengguna dan berwarna kuning untuk pemain lawan. *Source code* untuk menyusun huruf adalah :

```
private void searchWord(int startPos, int endPos, String searchedWord,
String[] letter) {
    if (startPos == -1 || endPos == -1) return;
    StringBuilder sb = new StringBuilder();
    ArrayList<Integer> grid;
    int gridSize = (int) Math.sqrt(letter.length);
    if (!isFoundWord()) {
        if (gameMode.equals("match")) {
            JSONObject data = new JSONObject();
            try {
                data.put("pos1", startPos);
                data.put("pos2", endPos);
            } catch (JSONException e) {
                e.printStackTrace();
            }
            mSocket.emit("player answer", data);
        }
        grid = searchWordGridBuilder(startPos, endPos, gridSize);
        for (int i = 0; i < grid.size(); i++) {
            sb.append(gameLetter[grid.get(i)]);
        }
        if (isWordFound(searchedWord, sb.toString(), grid, "player")) {
            setFoundWord(true);
            int score = Integer.parseInt(plScoreText.getText().toString());
            score = score + 1;
            eventText.setText("Kamu berhasil menemukan jawaban yang
dicari : "+answer);
            plScoreText.setText(String.valueOf(score));
            waitingDialog.show();
        }
    }
}
```

Source code di atas merupakan fungsi untuk menyusun huruf yang terbentuk ketika pemain mencari jawaban di dalam papan permainan. Kode *mSocket.emit("player answer", data)* digunakan untuk mengirim data posisi jawaban yang dijawab ke pada pemain lawan jika mode permainan bukan mode "SINGLE". Fungsi *sb.append(gameLetter[grid.get(i)])* berfungsi untuk menyusun huruf dari posisi *grid* pada papan permainan menjadi berbentuk *String* yang kemudian akan diperiksa apakah jawaban tersebut benar oleh fungsi *isWordFound*. Jika jawaban benar, susunan huruf tersebut berubah warna menjadi berwarna hijau untuk pengguna dan berwarna biru untuk pemain lawan—seperti yang terlihat pada Gambar 9-10, kemudian permainan dilanjutkan ke sesi berikutnya. Jika jawaban salah, susunan huruf tetap berwarna merah selama sepertiga detik kemudian kembali ke warna semula.



Gambar 9 Tampilan pemain mencari jawaban



Gambar 10 Tampilan pemain menemukan jawaban

Jika tidak ada pemain yang mampu menjawab dengan benar sebelum jangka waktu sesi menjawab habis, sistem komputer akan mencari jawaban tersebut menggunakan metode runut-balik(*backtracking*). Perhatikan *source code* berikut:

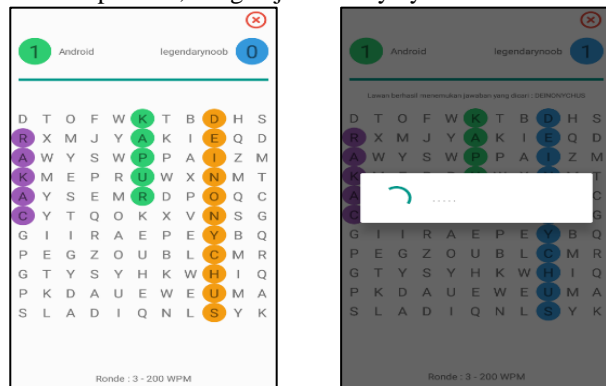
```
private void searchTheAnswer(String answer){
    final ProgressDialog dialog = new ProgressDialog(GameActivity.this);
    dialog.setMessage("Please wait...");
    String[] orientation = {"horizontal", "horizontalBack", "vertical",
        "verticalBack", "northEast", "southEast",
        "northWest", "southWest"};

    if (dialog == null || !dialog.isShowing()) {
        if(!GameActivity.this.isFinishing()){
            dialog.show();
        }
    }
    int x = 0;
    boolean found = false;
    while (!found) {
        if (!(x < 0 || x >= gameLetter.length)){
            if (answer.substring(0,1).equals(gameLetter[x])){
                for (String anOrientation : orientation) {
                    if (backtrackingSearch(answer, x, anOrientation)) {
                        dialog.dismiss();
                        found = true;
                        break;
                    }
                }
            }
            x++;
        }
    }
}
```

Langkah dari proses *backtracking* dari kode di atas yaitu selama kata belum ketemu, untuk setiap huruf pada papan permainan, lakukan pencarian huruf pertama dari jawaban di papan permainan dengan menggunakan kode *if(answer.substring(0,1).equals(gameLetter[x]))*. Kemudian untuk setiap kemungkinan posisi, lakukan pencarian untuk setiap huruf berikutnya dengan untuk setiap huruf ke-n harus sama dengan huruf ke-n dari jawaban yang dicari. Apabila huruf tidak sama, pencarian pada posisi tersebut dihentikan dan dilanjutkan dengan pencarian pada posisi selanjutnya. Apabila dalam pencarian ditemukan kata yang sesuai dengan jawaban, proses pencarian dihentikan menggunakan kode *break*.

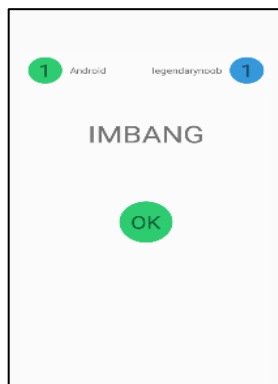
Pada contoh kasus ini, jawaban dari pertanyaan pertama adalah **KAPUR**. Hasil pencarian yang dilakukan oleh sistem komputer ditampilkan dengan warna ungu seperti yang terlihat pada Gambar 11 Setelah sesi menjawab pertanyaan pertama selesai, permainan dilanjutkan dengan sesi pertanyaan kedua. Kemudian dilanjutkan sesi

menjawab pertanyaan kedua. Begitu seterusnya hingga sesi menjawab pertanyaan ketiga. Pada contoh kasus ini, pertanyaan kedua yaitu: "Ciri khas dari dromaesauridae", dengan jawaban yang benar adalah **CAKAR**. Kemudian pertanyaan ketiga adalah: "Salah satu kerabat Utahraptor ...", dengan jawabannya yaitu **DEINONYCHUS**.



Gambar 11 Tampilan mencari jawaban pertanyaan dan Tampilan lawan menemukan jawaban pertanyaan ketiga

Setelah permainan berakhir, pemain akan dialihkan menuju tampilan skor seperti pada Gambar 12



Gambar 12 Tampilan skor

Simpulan

Aplikasi permainan ini sebagai alternatif yang dapat digunakan untuk bermain sambil belajar terutama pembelajaran atau membaca suatu kata. Telah dibuat aplikasi permainan adu membaca cepat berbasis android dengan menggabungkan teknik membaca cepat metode *scanning* dengan permainan cari kata (*word search puzzle*) yang dapat dimainkan dua orang secara bersama-sama. Proses implementasi dari aplikasi ini dibangun dengan menggunakan library Socket.IO sehingga mampu berjalan secara real time data-data terutama data pemain disimpan dengan menggunakan database MongoDB.

Daftar Pustaka

- Firdaus, M., & Nugroho, H. W. (2015). Rancang Bangun Game Edukasi Asah Otak Anak Berbasis Android Menggunakan Aplikasi Android Menggunakan Aplikasi. *Konvergensi*, 1-10.
- Jasson. (2009). *Role Playing Game (RPG) Maker*. Yogyakarta: Penerbit Andi, Yogyakarta.
- Munir, R. (2009). *Diktat Kuliah IF3051 Strategi Algoritma*. Bandung: Institut Teknologi Bandung.
- Nurhadi. (2005). *Bagaimana Meningkatkan Kemampuan Membaca*. Bandung: Sinar Baru Algensindo.
- P, S. F., Apriani, A., & v. (2014). *Rancang Bangun Aplikasi Edukasi "Ceria" Berbasis Android Sebagai Media Pembelajaran Untuk Anak Usia Dini*. Palembang: STMIK GI MDP .
- Putra, D. W., Nugroho, A. P., & Puspitarini, E. W. (2016). Game Edukasi Berbasis Android Sebagai Media Pembelajaran Untuk Anak Usia Dini. *JIMP - Jurnal Informatika Merdeka Pasuruan*, 46-58.
- Sibero, I. C. (2009). *Langkah Mudah Membuat Game 3D*. Yogyakarta: MediaKom.