

APLIKASI PEMBELAJARAN DASAR BAHASA JEPANG BERBASIS ANDROID

Aji Pujo Laksono¹, Maimunah²

¹Jurusan Teknik Informatika STMIK Nusa Mandiri, ²Program Studi Teknik Komputer, Universitas Islam 45
¹Jl. Jatiwaringin Raya Jakarta Timur, ²Jl. Cut Meutia 83 Bekasi

Email : ajipujolaksono@gmail.com

ABSTRACT

Utilization of smartphone technology can help people in learning a language with Android-based applications. Japan is one country in Asia that has a unique and interesting culture including language culture. One of the fundamental problems in Japanese is a need to study Japanese script, which has many types of characters are used: Hiragana, Katakana and Kanji. Besides Japanese characters also have ordinances order written on each character. Japanese language learning mobile applicatiois very easy for someone to learn Japanese. In this research, the Japanese language learning applications designed using the Android 4.0.1 (Ice Cream Sandwith). The method used in the search vocabulary of the Japanese language support using brute force algorithm. The results of this research is Japanese learning application that provides an introduction material katakana, hiragana, using voice and vocabulary quiz question.

Keywords: Android Apps, Japanese, learning

ABSTRAK

Pemanfaatan teknologi smartphone dapat membantu manusia dalam mempelajari suatu bahasa dengan aplikasi berbasis Android. Jepang merupakan salah satu negara di Asia yang memiliki kebudayaan yang unik dan menarik termasuk kebudayaan bahasa. Salah satu permasalahan mendasar dalam bahasa Jepang adalah perlunya mempelajari aksara-aksara Jepang yang mempunyai banyak jenis aksara yang digunakan, antara lain *Hiragana*, *Katakana* dan *Kanji*. Selain itu aksara Jepang juga memiliki tata cara urutan penulisan pada tiap hurufnya. Aplikasi berbasis mobile tentang pembelajaran tentang bahasa Jepang sangat memudahkan seseorang dalam mempelajari bahasa Jepang. Dalam penelitian ini dirancang aplikasi pembelajaran bahasa Jepang menggunakan menggunakan Android 4.0.1 (*Ice Cream Sandwith*). Metode yang digunakan dalam pencarian kosa kata bahasa Jepang menggunakan algoritma brute force. Hasil penelitian ini berupa aplikasi pembelajaran dasar bahasa Jepang yang menyediakan materi pengenalan huruf katakana, huruf hiragana, kosakata menggunakan suara dan soal kuis.

Kata Kunci: Aplikasi Android, bahasa Jepang, pembelajaran

1. Pendahuluan

Negara Jepang merupakan satu dari beberapa negara di Asia yang mempunyai budaya yang unik dan menarik. Negara Jepang telah menyebarkan budayanya ke seluruh dunia salah satunya yaitu negara Indonesia melalui produk-produk budaya populer seperti *Anime* (video animasi Jepang), *manga* (komik Jepang), dan makanan khas Jepang (Mardhana:2011). Selain budaya, perkembangan teknologi, pendidikan, dan bisnis membuat banyak

masyarakat Indonesia yang tertarik untuk belajar atau berbisnis dengan orang Jepang. Hal ini menjadikan bahasa Jepang salah satu bahasa asing yang banyak dipelajari di negara Indonesia.

Salah satu permasalahan mendasar dalam bahasa Jepang adalah perlunya mempelajari aksara-aksara Jepang. Banyak orang menganggap bahwa bahasa Jepang merupakan bahasa yang sulit dipelajari. Hal ini disebabkan oleh banyaknya jenis aksara yang digunakan, antara lain *Hiragana*,

Katakana dan *Kanji*, jika dibandingkan dengan alphabet yang digunakan dalam bahasa Indonesia yang hanya terdiri atas 26 huruf saja. Aksara Jepang juga memiliki tata cara urutan penulisan pada tiap hurufnya, sama seperti aksara Mandarin yang merupakan asal dari aksara Jepang.

Menurut Sumarlin, dkk (2015:1) “Dalam proses untuk mempelajari aksara Jepang, keberadaan suatu perangkat ajar dibutuhkan untuk mendukung suatu proses belajar. Salah satu aspek yang membuat sulit dalam mempelajari bahasa Jepang terdapat pada sistem penulisan. Salah satu perangkat elektronik yang dikira efektif sebagai media aplikasi edukasi adalah *mobile device*. Aplikasi pembelajaran aksara Jepang sebelumnya memiliki permasalahan seperti kurangnya pengucapan kosakata yang detail dan soal latihan untuk mengetahui seberapa besar pengetahuan pembelajaran dalam aplikasi tersebut”

Kemunculan berbagai perangkat elektronik dengan diiringi perkembangan teknologi dapat dimanfaatkan untuk mengembangkan aplikasi dalam membantu proses pembelajaran, seperti *game* edukasi pada perangkat elektronik. Perangkat elektronik yang dinilai efektif sebagai media pembelajaran adalah perangkat *mobile*, salah satu contohnya yaitu *smartphone*. *Smartphone* mampu menjadi perangkat yang populer dan banyak dibutuhkan oleh semua kalangan dari orangtua maupun anak-anak. Maka untuk mengolah sumber daya perangkat keras yang dimiliki oleh *smartphone* perlu menggunakan sistem

operasi. Salah satu sistem operasi *smartphone* adalah android.

Ketersediaan berbagai fitur pada android yang memungkinkan pengguna dalam mengembangkan aplikasi, mendasari dalam pembuatan aplikasi edukasi pembelajaran bahasa Jepang ini berbasis Android. Maksud dari penelitian ini adalah membangun suatu rancangan aplikasi berbasis android sebagai media pembelajaran dasar bahasa Jepang. Adapun ruang lingkup dalam penelitian ini membahas tentang aplikasi pembelajaran dasar bahasa Jepang dengan memberi batasan yang meliputi pengenalan huruf hiragana, katakana, dan kosakata dalam bahasa Jepang dengan ditambahkan suara pengucapannya, dan kuis serta menerapkan metode string *matching* dengan algoritma *brute force* dalam menu pencarian di dalam kosakata.

2. Bahan dan Metode Penelitian

2.1. Bahan

Adapun perangkat lunak yang dibutuhkan dalam perancangan aplikasi ini antara lain:

1. Java Development Kit
2. Android Studio sebagai IDE (*Interface Development Kit*)
3. Android SDK (*Software Development Kit*)
4. Android ADT (*Android Development Tools*) sebagai penghubung IDE Android Studio dengan Android SDK.
5. Minimal Android OS 4.0 (Froyo) sampai dengan Android versi 6.0 (Marshmallow)

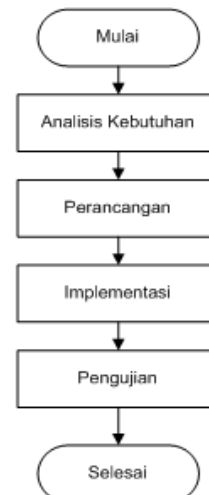
2.2. Metode Penelitian

Algoritma yang digunakan pada aplikasi ini adalah algoritma *String Matching* yang merupakan salah satu algoritma yang digunakan untuk mempercepat proses pencarian kata. Algoritma *string matching* telah sering digunakan pada proses pencocokan string berdasarkan persamaan teks data yaitu *Brute Force*. Algoritma *brute force* memecahkan masalah dengan sangat sederhana, langsung dan dengan cara yang jelas (*obvious way*) langsung ke pusat permasalahan. *Brute Force* merupakan algoritma pencarian string termudah. Dengan asumsi bahwa teks berada di dalam array $T[1..n]$ dan pattern berada di dalam array $P[1..M]$ maka algoritma *Brute Force* pencocokan *string* adalah sebagai berikut:

1. Mula-mula *pattern P* dicocokkan pada awal teks *T*
 2. Dengan bergerak dari kiri ke kanan, bandingkan setiap karakter di dalam *pattern P* dengan karakter yang bersesuaian di dalam teks *T* sampai:
 - a. Semua karakter yang dibandingkan cocok atau sama (pencarian berhasil) atau
 - b. Dijumpai sebuah ketidakcocokan karakter (pencarian belum berhasil)
 3. Bila *pattern P* belum ditemukan kecocokannya dan teks *T* belum habis, geser *pattern P* satu karakter ke kanan dan ulangi langkah 2
- Persoalan pencocokan string dapat dirumuskan sebagai berikut:

1. Teks (*text*) yaitu (*long*) string yang panjangnya n karakter
2. Pattern yaitu string dengan panjang m karakter ($m < n$) yang akan dicari di dalam teks

Tahapan penelitian dalam pembuatan aplikasi ini seperti dalam Gambar 1 terdiri dari 4 tahap, yaitu analisis kebutuhan, perancangan, implementasi dan pengujian. Dalam tahap analisis kebutuhan dilakukan penelitian dan pengamatan tentang apa yang menjadi permasalahan dan kebutuhan dalam pembelajaran bahasa Jepang. Dari permasalahan dan kebutuhan tersebut dapat diperoleh solusi dari permasalahan dan membuat gambaran umum dari aplikasi yang akan dibuat.



Gambar 1. Tahapan penelitian

Tahap perancangan merupakan pengembangan dari gambaran umum sistem. Dalam tahap perancangan dijelaskan lebih detail tentang isi dari aplikasi yang dibuat yaitu dengan membuat diagram *Unified Modelling Language* (UML) yang meliputi *use case*, *activity diagram*, *sequence diagram* dan *class diagram* serta membuat desain *input* dan *output*. Setelah tahap

perancangan selesai maka dilakukan tahap implementasi yaitu menerjemahkan desain ke dalam *source code* berbasis Android.

3. Hasil dan Pembahasan

3.1. Hasil

A. Analisis kebutuhan

Adapun masalah yang akan diteliti dalam pembuatan aplikasi pembelajaran dasar bahasa jepang adalah sebagai berikut:

1. Belajar bahasa jepang dalam bentuk buku memiliki kekurangan karena sifat dari kertas yang rentan rusak serta warna yang mudah memudar.

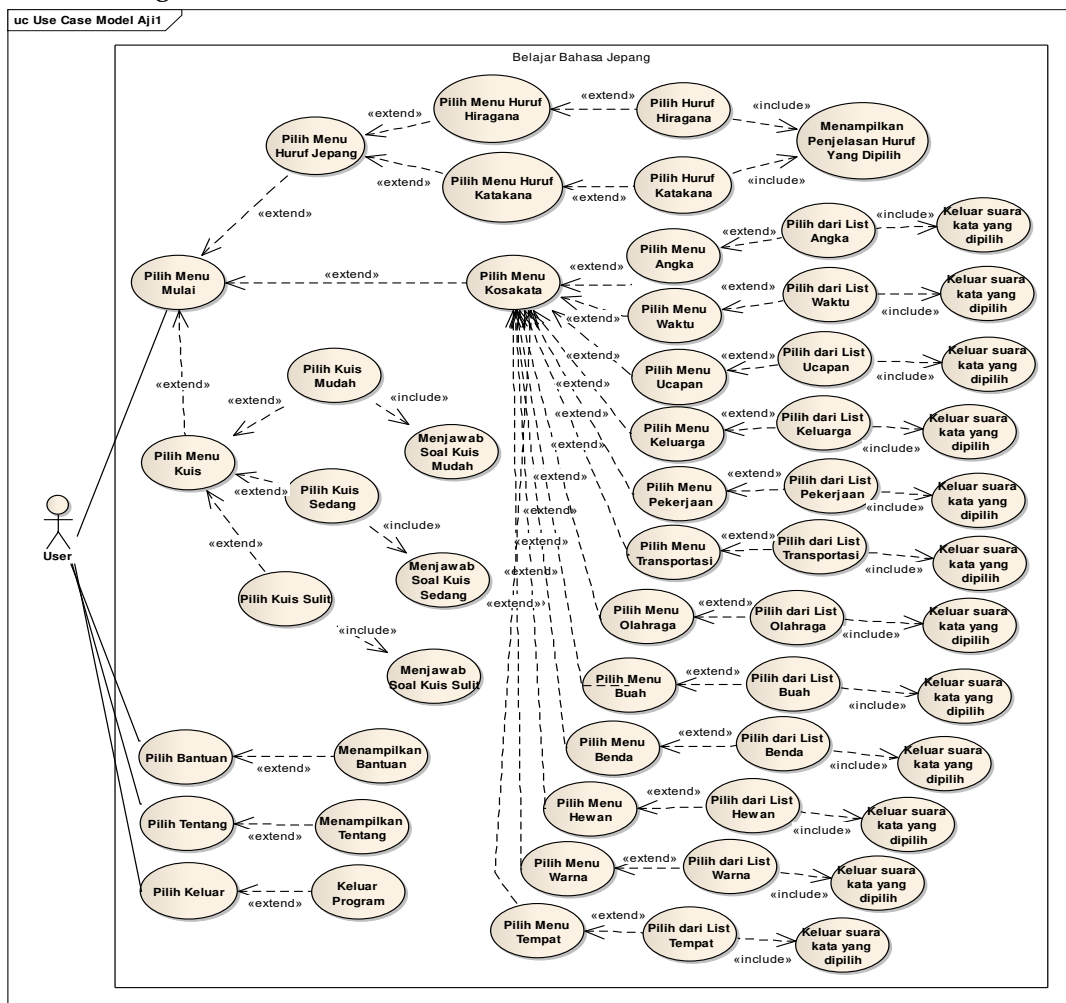
2. Sulit belajar bahasa jepang dengan hanya mengandalkan tulisan atau dalam bentuk text, karena lebih mudah dengan bantuan audio atau suara sehingga lebih jelas pelafalanya.

3. Lebih efisien waktu dan tempat apabila pembelajaran bahasa jepang dikemas dalam bentuk aplikasi yang lebih dekat dengan pengguna seperti *smartphone* dengan operasi sistem android.

B. Perancangan

Pada tahap perancangan dilakukan dengan membuat *Unified Modelling Language (UML)* yang meliputi:

1. Use Case Diagram



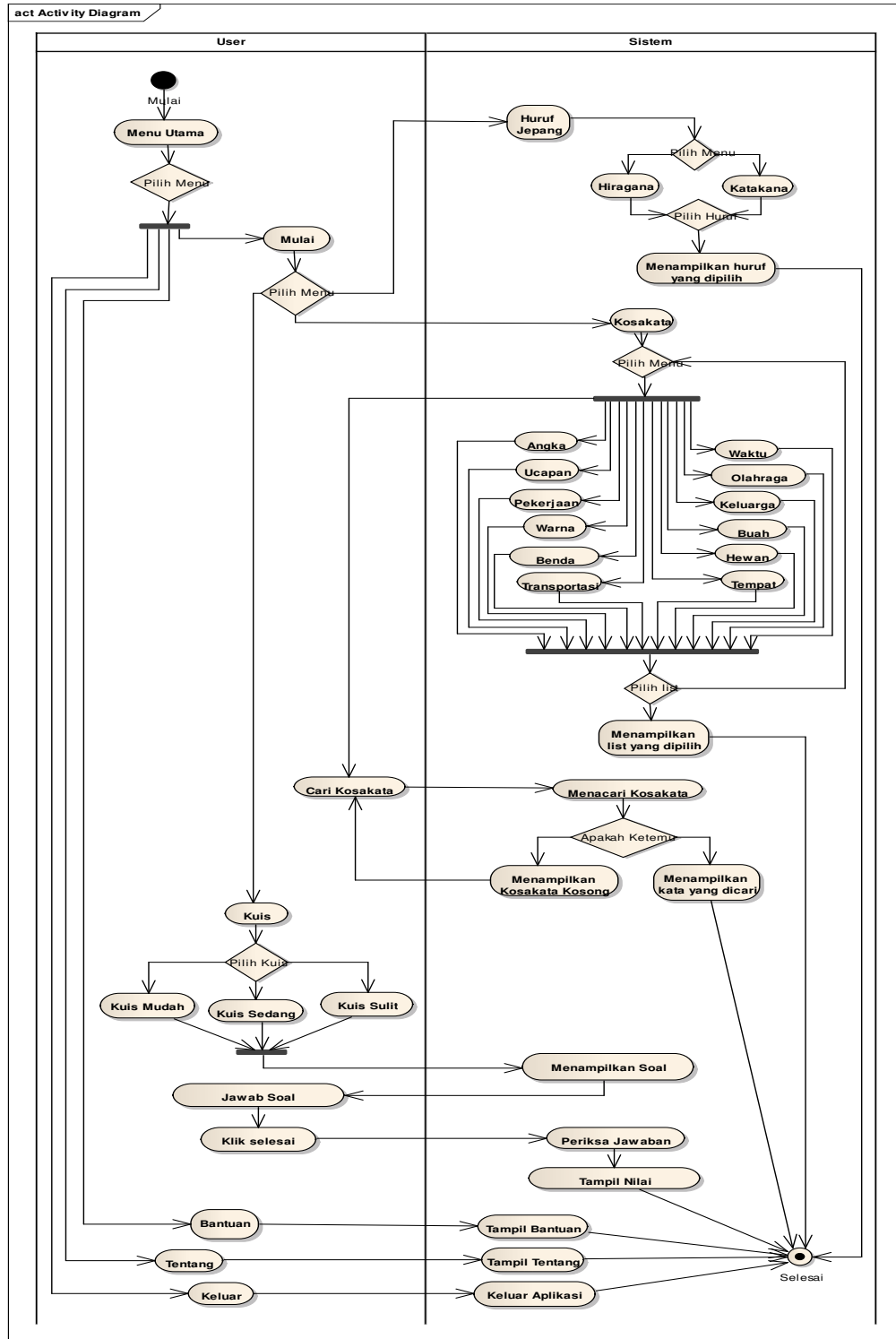
Gambar 2. Use case diagram

2. Activity Diagram

Activity diagram menggambarkan diagram alir aktivitas dalam sistem yang sedang dirancang. Activity diagram untuk aplikasi yang dirancang seperti dalam Gambar 3.

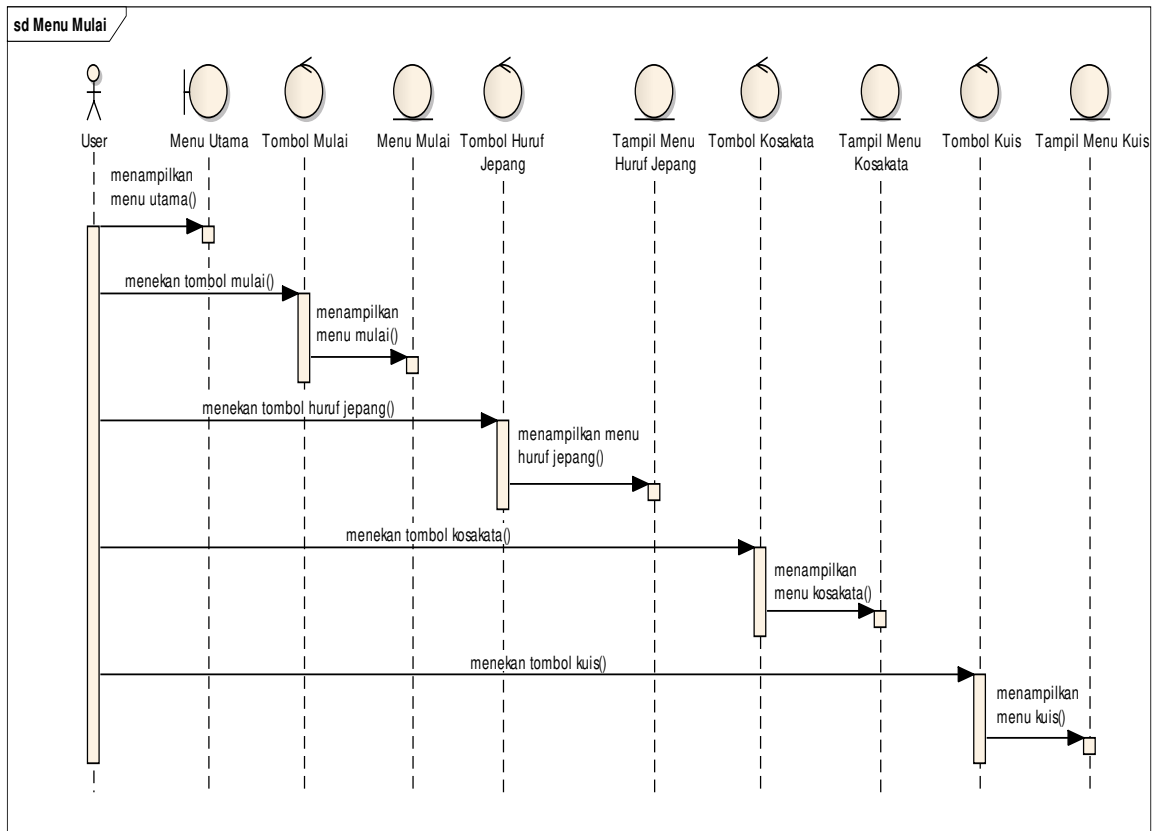
3. Sequence Diagram

Sequence diagram merupakan diagram yang menjelaskan tentang urutan yang terdapat pada sistem. Aplikasi kamus bahasa Betawi mempunyai diagram sequence seperti dalam 4,5 dan 6.



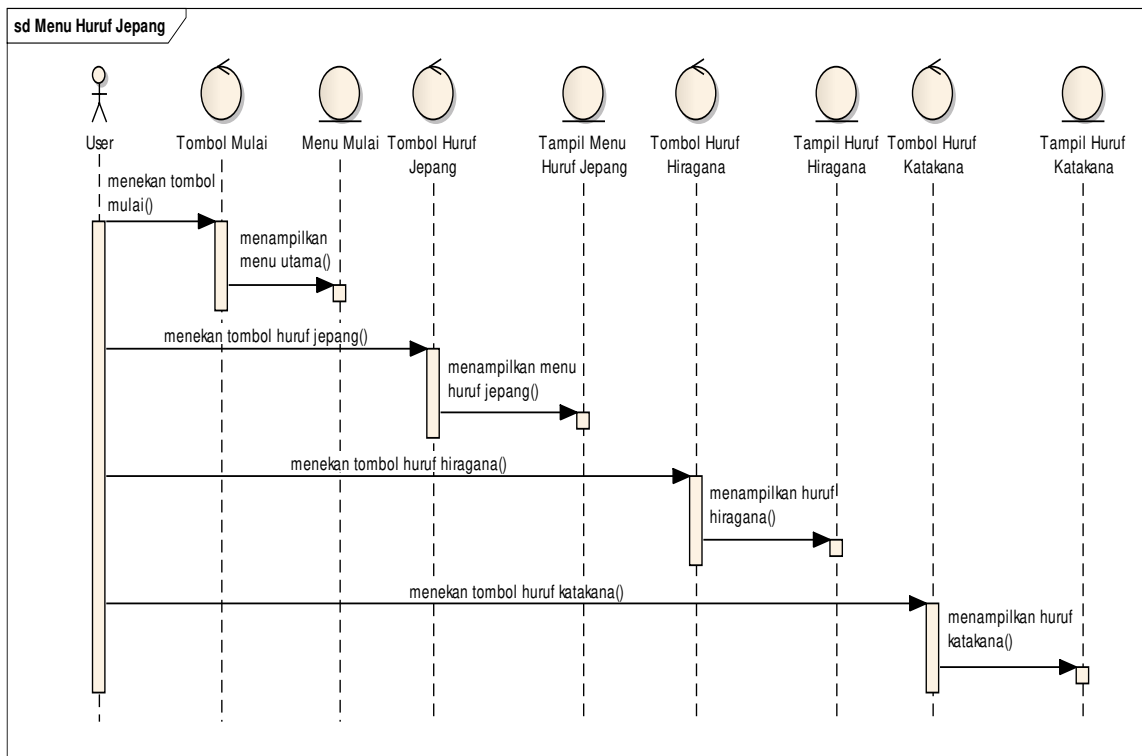
Gambar 3. Activity diagram

a) Sequence Diagram Menu Mulai



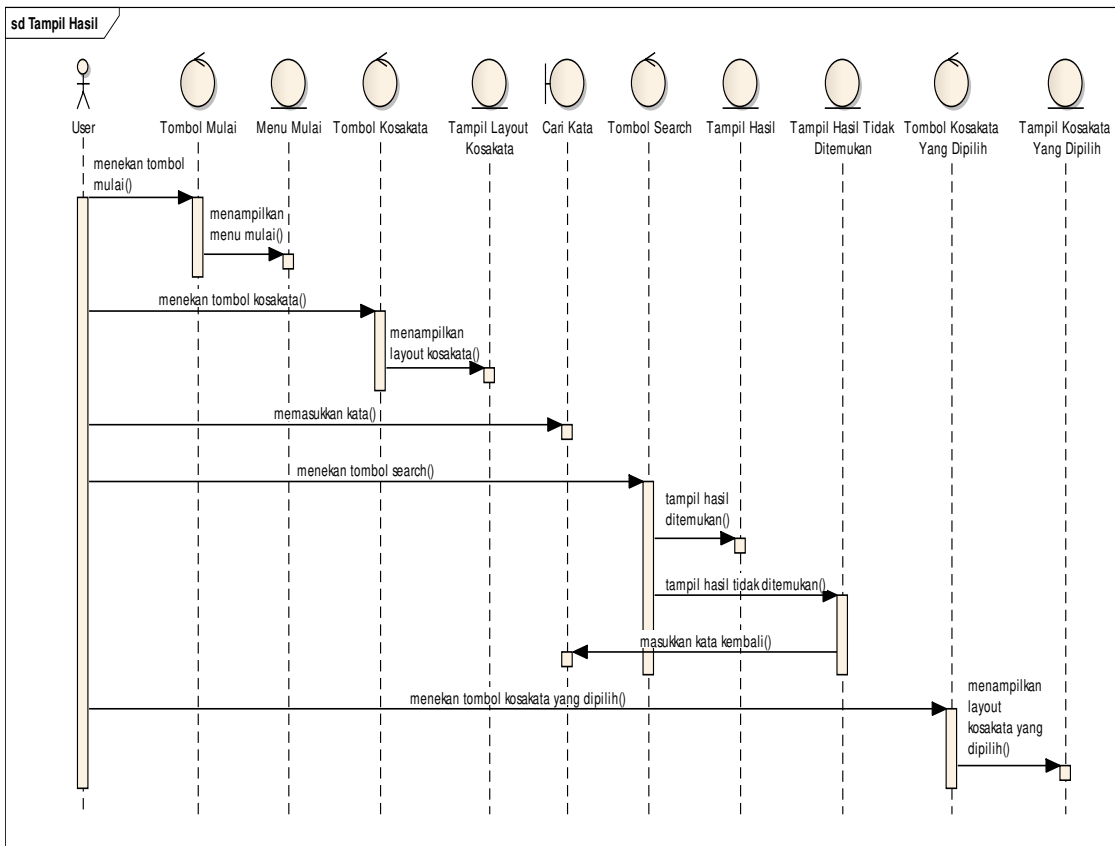
Gambar 4. Sequence diagram menu mulai

b) Sequence Diagram Menu Huruf Jepang



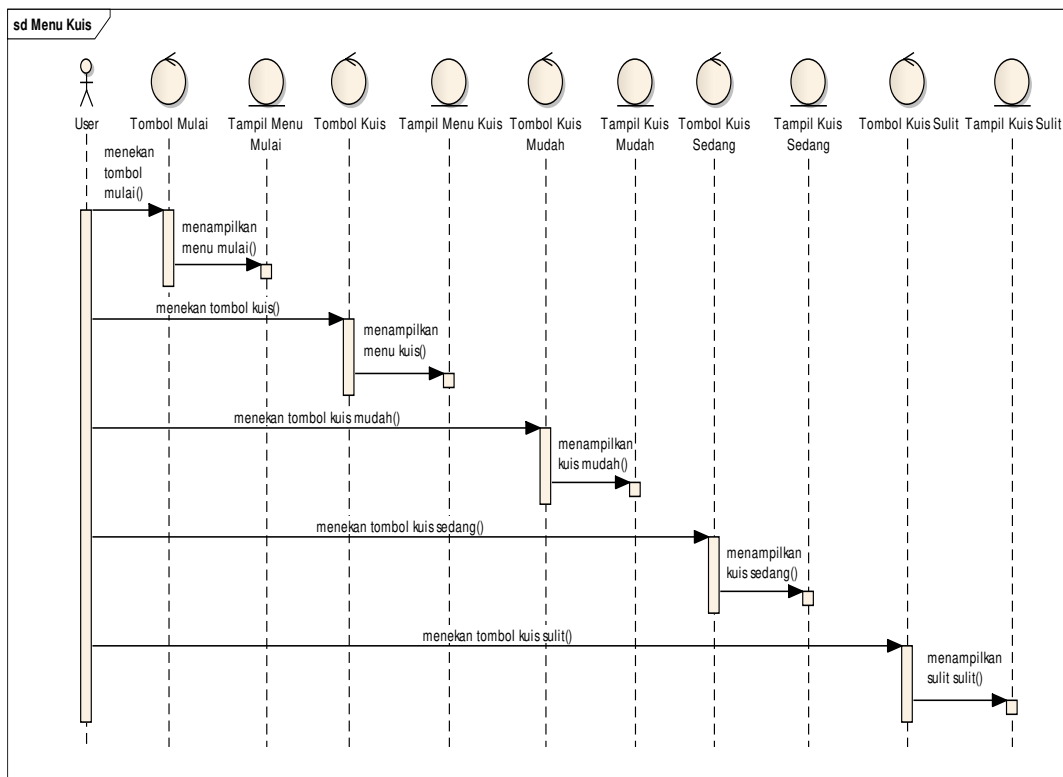
Gambar 5. Sequence diagram menu huruf Jepang

c) Sequence Diagram Menu Kosakata



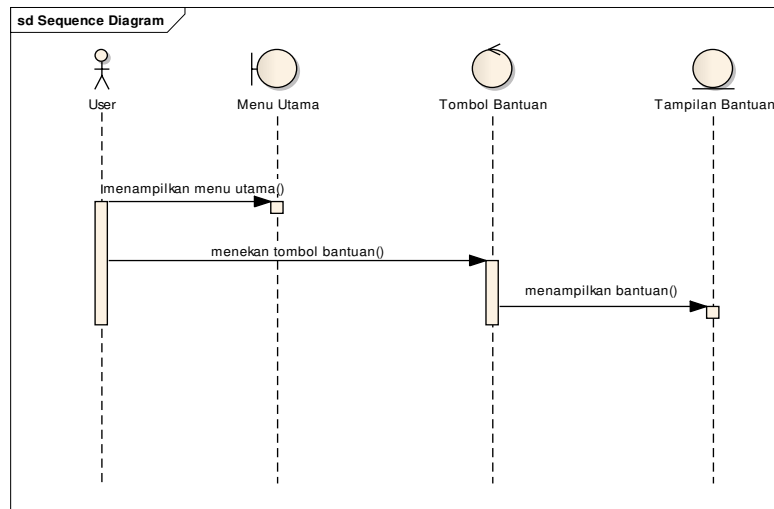
Gambar 6. Sequence diagram menu kosakata

d) Sequence Diagram Menu Kuis



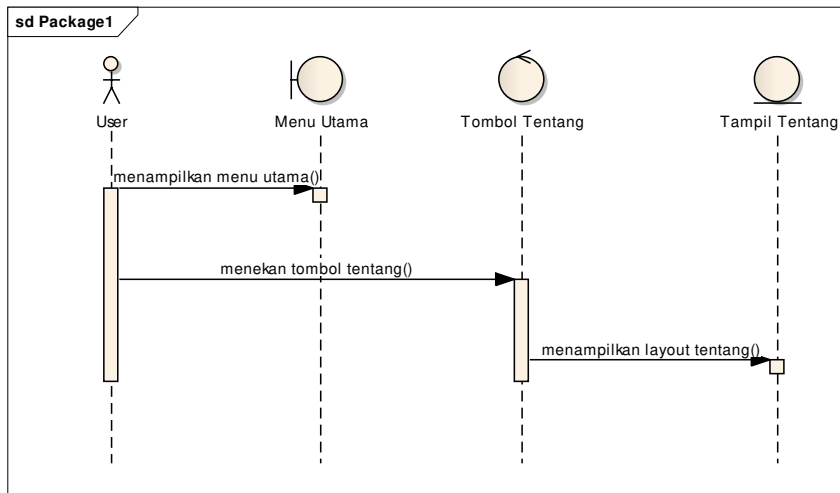
Gambar 7. Sequence diagram menu kuis

e) *Sequence Diagram Menu Bantuan*



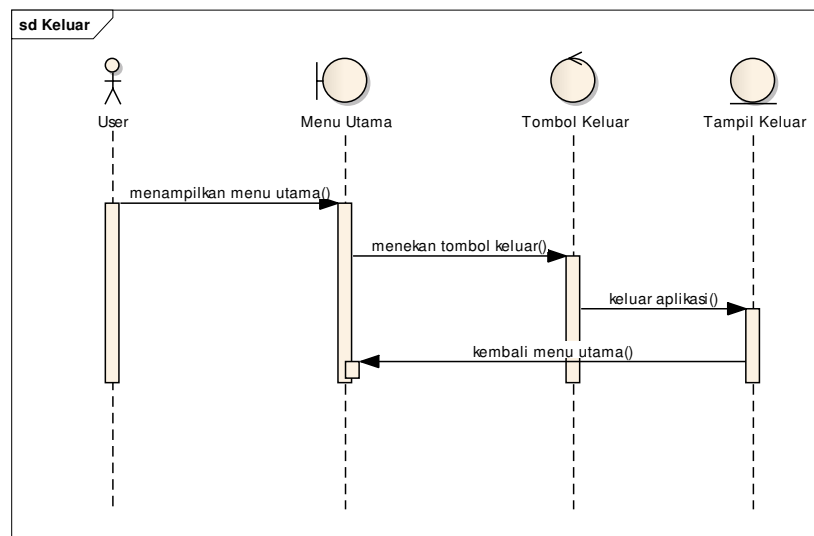
Gambar 8. *Sequence diagram* menu bantuan

f) *Sequence Diagram Menu Tentang*



Gambar 9. *Sequence diagram* menu tentang

g) *Sequence Diagram Menu Keluar*



Gambar 10. *Sequence diagram* menu keluar



Gambar 15. Tampilan menu huruf Jepang



Gambar 18. Tampilan menu kosakata



Gambar 16. Tampilan menu huruf Hiragana



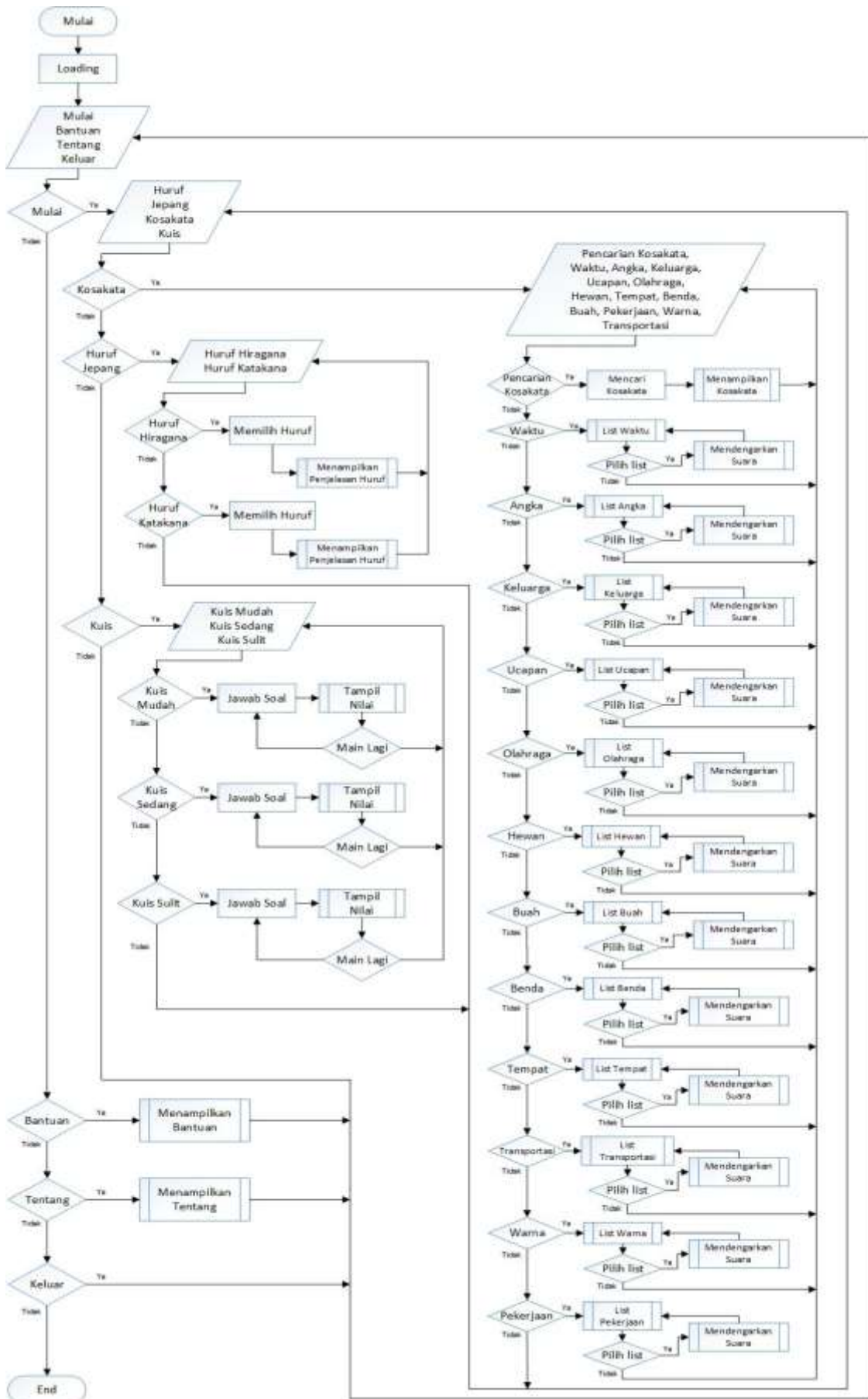
Gambar 19. Tampilan menu kuis



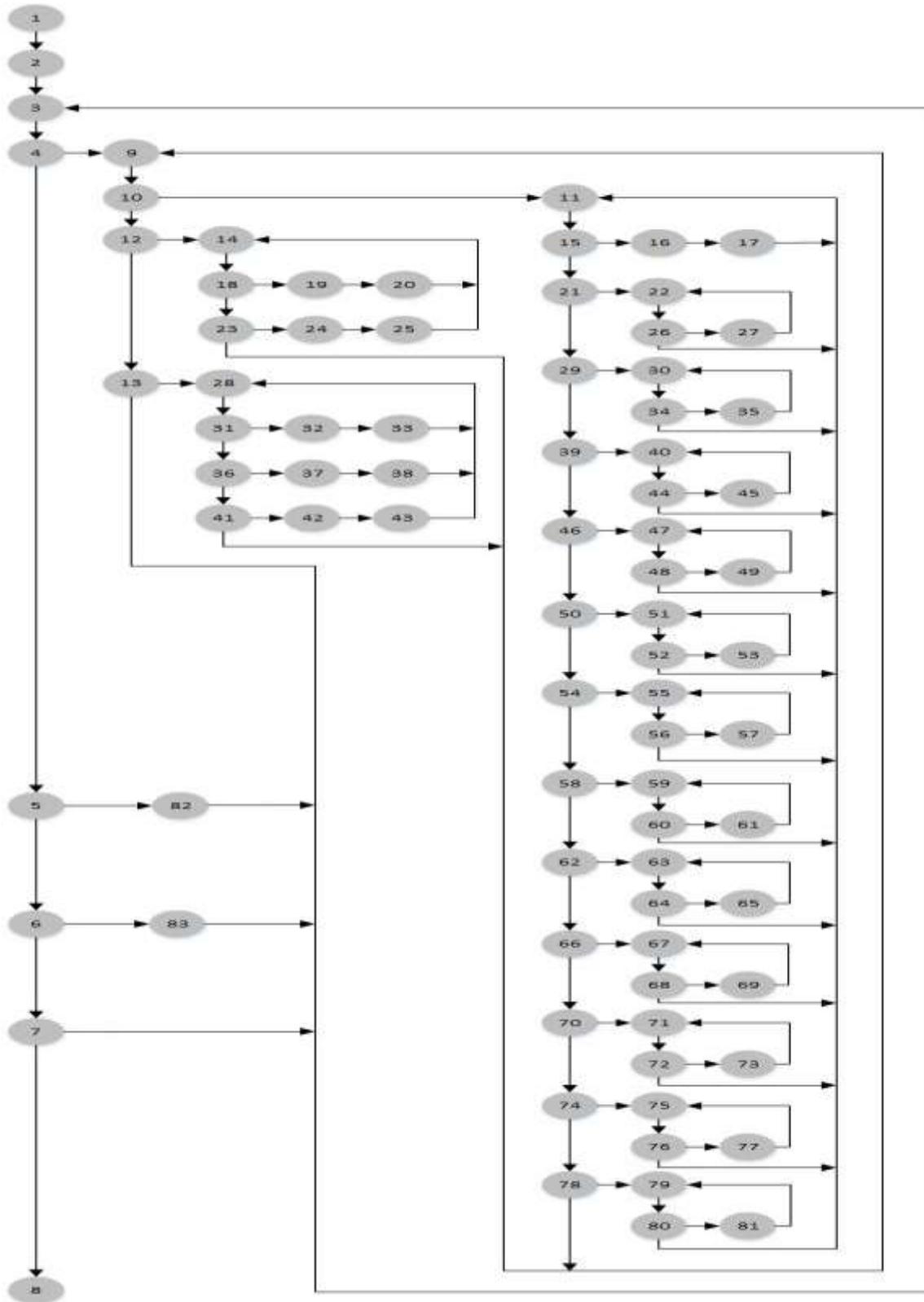
Gambar 17. Tampilan menu huruf Katakana

D. Pengujian

Pengujian dari aplikasi yang telah dirancang terdiri dari *white box testing* dan *black box testing*. Pengujian dengan menggunakan *white box* menguji perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi-fungsi, masukan, dan keluaran yang sesuai dengan spesifikasi kebutuhan. Pengujian kotak putih dilakukan dengan memeriksa logika dari kode program. Flowchart aplikasi seperti Gambar 20 dan diagram alir aplikasi kamus bahasa Betawi seperti dalam Gambar 21.



Gambar 20. Flowchart aplikasi



Gambar 21. *White box testing*

Pengukuran kompleksitas siklomatis *logis* suatu program pada grafik alir diatas dapat diperoleh dengan perhitungan: $V(G) = E - N + 2$, dimana:

- a. E = Merupakan jumlah tepi (*Edge*) grafik alir yang ditandai dengan gambar anak panah.

- b. N = Jumlah simpul (*Node*) grafik alir yang ditandakan dengan gambar lingkaran.

Sehingga kompleksitas siklomatis yang dihasilkan adalah $V(G) = 115 - 83 + 2$, maka $V(G) = 34$.

Berdasarkan hasil perhitungan kompleksitas siklomatis, pengujian menggunakan jalur independen yang mana setiap jalur melalui program yang memperkenalkan satu set baru pernyataan pemrosesan. Ketika dinyatakan dalam grafik alur, jalur independen bergerak sepanjang satu sisi yang belum di lalui. Hasil pengujian satu set jalur untuk grafik alir yang dihasilkan dari jalur *independent* adalah sebagai berikut:

- 1) 1-2-3-4-9-10-12-14-18-19-20-14-12-13-3-4-5-6-7-8
- 2) 1-2-3-4-9-10-12-14-18-23-24-25-14-12-13-3-4-5-6-7-8
- 3) 1-2-3-4-9-10-12-14-18-23-9-3-4-5-6-7-8
- 4) 1-2-3-4-9-10-12-13-28-31-32-33-28-9-3-4-5-6-7-8
- 5) 1-2-3-4-9-10-12-13-28-31-36-37-38-28-9-3-4-5-6-7-8
- 6) 1-2-3-4-9-10-12-13-28-31-36-41-42-43-28-9-3-4-5-6-7-8
- 7) 1-2-3-4-9-10-12-13-28-31-36-41-9-3-4-5-6-7-8
- 8) 1-2-3-4-9-10-11-15-16-17-11-9-3-4-5-6-7-8
- 9) 1-2-3-4-9-10-11-15-21-22-26-27-22-11-9-3-4-5-6-7-8
- 10) 1-2-3-4-9-10-11-15-21-22-26-11-9-3-4-5-6-7-8
- 11) 1-2-3-4-9-10-11-15-21-29-30-34-35-30-11-9-3-4-5-6-7-8
- 12) 1-2-3-4-9-10-11-15-21-29-30-34-11-9-3-4-5-6-7-8
- 13) 1-2-3-4-9-10-11-15-21-29-39-40-44-45-40-11-9-3-4-5-6-7-8
- 14) 1-2-3-4-9-10-11-15-21-29-39-40-44-11-9-3-4-5-6-7-8
- 15) 1-2-3-4-9-10-11-15-21-29-39-46-47-48-49-47-11-9-3-4-5-6-7-8
- 16) 1-2-3-4-9-10-11-15-21-29-39-46-47-48-11-9-3-4-5-6-7-8
- 17) 1-2-3-4-9-10-11-15-21-29-39-46-50-51-52-53-51-11-9-3-4-5-6-7-8
- 18) 1-2-3-4-9-10-11-15-21-29-39-46-50-51-52-11-9-3-4-5-6-7-8
- 19) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-55-56-57-55-11-9-3-4-5-6-7-8
- 20) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-55-56-11-9-3-4-5-6-7-8
- 21) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-59-60-61-59-11-9-3-4-5-6-7-8
- 22) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-59-60-11-9-3-4-5-6-7-8
- 23) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-62-63-64-65-63-11-9-3-4-5-6-7-8
- 24) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-62-63-64-11-9-3-4-5-6-7-8
- 25) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-62-66-67-68-69-67-11-9-3-4-5-6-7-8
- 26) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-62-66-67-68-11-9-3-4-5-6-7-8
- 27) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-62-66-70-71-72-73-71-11-9-3-4-5-6-7-8
- 28) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-62-66-70-71-72-11-9-3-4-5-6-7-8

- 29) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-62-66-70-74-75-76-77-75-11-9-3-4-5-6-7-8
- 30) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-62-66-70-74-75-76-11-9-3-4-5-6-7-8
- 31) 1-2-3-4-9-10-11-15-21-29-39-46-50-54-58-62-66-70-74-78-79-80-81-79-11-9-3-4-5-6-7-8
- 32) 1-2-3-4-5-82-3-4-5-6-7-8
- 33) 1-2-3-4-5-6-83-3-4-5-6-7-8

- 34) 1-2-3-4-5-6-7-3-4-5-6-7-8

Pengujian *Black Box* bertujuan untuk menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Hasil pengujian menggunakan black box testing seperti dalam table 1.

Tabel 1. Tabel Black Box Testing

Skenario uji	Test Case	Hasil yang diharapkan	Keterangan
Tampil Menu Utama	Menampilkan Menu Utama	Tampil Menu Utama	Sesuai
Memilih Mulai	Menampilkan Menu Mulai	Tampil Menu Mulai	Sesuai
Memilih Huruf Jepang	Menampilkan Menu Huruf Jepang	Tampil Menu Huruf Jepang	Sesuai
Memilih huruf hiragana	Menampilkan daftar huruf hiragana	Tampil daftar huruf hiragana	Sesuai
Memilih huruf katakana	Menampilkan daftar huruf katakana	Tampil daftar huruf katakana	Sesuai
Memilih salah satu huruf dari daftar huruf hiragana	Menampilkan cara penulisan dari huruf hiragana yang dipilih	Tampil layout cara penulisan dari huruf yang dipilih	Sesuai
Memilih salah satu huruf dari daftar huruf katakana	Menampilkan cara penulisan dari huruf katakana yang dipilih	Tampil layout cara penulisan dari huruf yang dipilih	Sesuai
Memilih Kosakata	Menampilkan Menu Kosakata	Tampil Menu Kosakata	Sesuai
Mencari kosakata	Menampilkan kosakata yang dicari	Tampil kosakata yang dicari	Sesuai
Memilih salah satu daftar kosakata	Menampilkan list kosakata yang dipilih	Tampil list kosakata yang dipilih	Sesuai
Memilih Kuis	Menampilkan Menu Kuis	Tampil Menu Kuis	Sesuai
Memilih kuis mudah	Menampilkan soal kuis mudah	Tampil soal kuis mudah	Sesuai
Memilih kuis sedang	Menampilkan soal kuis sedang	Tampil soal kuis sedang	Sesuai
Memilih kuis sulit	Menampilkan soal kuis sulit	Tampil soal kuis sulit	Sesuai
Memilih bantuan	Menampilkan info bantuan	Tampil info bantuan	Sesuai
Memilih tentang	Menampilkan info tentang	Tampil info tentang	Sesuai
Memilih keluar	Keluar dari menu utama	Tampil halaman keluar	Sesuai

3.2 Pembahasan

Algoritma *brute force* yang digunakan dalam penelitian ini

memecahkan masalah dengan sangat sederhana, langsung dan dengan cara yang jelas. Pencocokkan karakter per

karakter *pattern* dengan karakter yang ada pada teks yang berkesesuaian mulai dari kiri ke kanan, sampai salah satu kondisi berikut terpenuhi yaitu karakter yang ada pada *pattern* dan pada teks yang dibandingkan tidak cocok dan semua karakter pada *pattern* cocok. Kemudian algoritma akan memberitahu penemuan di posisi ini. Algoritma kemudian terus menggeser *pattern* sebesar satu karakter ke kanan, dan mengulangi langkah ke-2 sampai kata yang berada diujung teks. Kelemahan Algoritma *Brute Force* yaitu lemah dalam sisi penggunaan memori pada komputer jika data yang diinginkan berada dalam titik tengah ke atas. *Brute Force* menjadi sangat efektif ketika data yang akan dicari berada pada titik tengah ke bawah. Aplikasi yang dirancang dilengkapi dengan pengucapan dalam bentuk suara sehingga memudahkan dalam pembelajaran bahasa Jepang.

4. Kesimpulan dan Saran

4.1. Kesimpulan

Dengan adanya aplikasi pembelajaran dasar bahasa Jepang berbasis android disertai dengan cara penulisan, fitur suara dan juga kuis yang akan lebih memaksimalkan dalam pembelajaran bahasa Jepang. Fitur kuis ini membuat pengguna dapat melihat seberapa jauh dalam pembelajaran yang diraih. Maka penulis dapat mengambil kesimpulan dari penulisan ini yang diharapkan dapat

bermanfaat dan berguna bagi masyarakat umum sebagai berikut:

1. Media android merupakan media yang cocok untuk membangun aplikasi pembelajaran dasar bahasa Jepang.
2. Aplikasi pembelajaran dasar bahasa Jepang berbasis android dapat digunakan sebagai media pembelajaran dasar bahasa Jepang yang efektif dan efisien.
3. Algoritma Brute Force dapat diterapkan pada aplikasi pembelajaran bahasa Jepang.

4.2. Saran

Aplikasi pembelajaran dasar bahasa Jepang berbasis android ini masih memiliki kekurangan, maka ada beberapa saran yang dapat diajukan untuk pengembangan dan penyempurnaan lebih lanjut, antara lain:

1. Menambahkan huruf Kanji sehingga menjadi lengkap pembelajaran huruf-huruf Jepang.
2. Penambahan jumlah kosakata sehingga lebih lengkap.

Daftar Pustaka

- Rosa dan Shalahuddin. 2015. *Rekayasa Perangkat Lunak*, Penerbit Informatika, Bandung
- Satyaputra, Alfa dan Aritonang, Eva, Maulina. 2014. *Beginning Android Programming With ADT Bundle*, Penerbit PT Elex Media Komputindo, Jakarta
- Safaat, Nazzarudin. 2014. *Pemogramman Aplikasi Mobile Smartphone dan Tablet PC*, Penerbit Informatika, Bandung
- Rama, Gede, Dewa, I. 2008. *Algoritma Pemogramman Java*. Bandung

Mesran. 2014. Implementasi Algoritma Brute Force Dalam Pencarian Data Katalog Buku Perpustakaan, ISSN: 2339-210X

Sumarlin, Zandy, Herlan dan Sutanto, Teguh. 2015. Aplikasi Pembelajaran

Aksara Jepang Berbasis Android, ISSN: 2338-137X

Yusa, Marthana, Made, I. 2013. Pengembangan Model Pembelajaran Huruf Hiragana Dalam Wujud CD Interaktif Untuk Siswa Kelas IV SD, ISSN: 2089-8673