

BAB II

LANDASAN TEORI

2.1 Tinjauan Terhadap Penelitian Terkait

Berikut beberapa hasil penelitian terkait sebelumnya yang menjadi referensi saya pada penelitian ini:

1. Kevin Angka Wijaya, Justinus Andjarwirawan, Lily Puspa Dewi dalam penelitian 2022 berjudul “Aplikasi Self Management untuk Mencatat Jadwal Kegiatan Dengan Speech to Text Menggunakan Google API Berbasis Android”. Pada penelitian ini permasalahan yang dihadapi yaitu Orang cenderung hanya mengandalkan ingatan dan asal mengatur jadwal kegiatan. Hal ini membuat orang banyak pikiran dan cenderung melupakan sesuatu. Dengan ini, Maka dibuatkan penelitian membuat aplikasi untuk penjadwalan menggunakan speech to text. Aplikasi tersebut memiliki fitur kalender, daftar kegiatan, catatan, dan speech to text. Speech to text menggunakan Google Speech API. Fitur kalender terhubung langsung dengan Google Calendar menggunakan Google Calendar API. Fitur lainnya disimpan pada database local berupa SQLite. Kemudian Aplikasi diuji dengan mengukur kecepatan pembuatan jadwal menggunakan stopwatch. Hasil yang didapatkan, jadwal dapat terbuat selama 10.426 detik menggunakan speech to text. Dengan menggunakan Google Speech API dalam pembuatan jadwal.

2. Yuri Rahmanto, Joni Alfian, Damayanti, Rohmat Indra Borman dalam penelitian 2021 berjudul “Penerapan Algoritma Sequential Search pada Aplikasi Kamus Bahasa Ilmiah Tumbuhan”. Permasalahan yang di hadapi pada penelitian ini yaitu untuk mempelajari istilah-istilah ilmiah maka membutuhkan kamus bahasa ilmiah. Akan tetapi kamus biasanya berbentuk buku yang tebal sehingga membuat para siswa yang akan belajar bahasa ilmiah menjadi tidak tertarik untuk mempelajarinya. Maka dibuatkan aplikasi untuk melakukan pencarian dengan metode sequential search dengan mencocokkan data secara beruntun dari seluruh data yang ada. Aplikasi yang dibangun memiliki fitur-fitur seperti daftar tumbuhan dengan nama ilmiah yang dilengkapi menu abjad A-Z, kategori tumbuhan, dan fitur pencarian untuk memudahkan pengguna dalam menemukan nama tumbuhan. Hasil pengujian kecepatan pencarian berdasarkan page load time dengan fungsi microtime menunjukkan bahwa algoritma sequential search dapat melakukan pencarian dengan cepat untuk data yang relatif terbatas atau tidak terlalu banyak.
3. Fitriah Halimah, Rita Purnamasari, Inang Wijayanto dalam penelitian 2017 berjudul “Analisis Sequential Searching Aplikasi Kesehatan Balita Dan Ibu Hamil Serta Pencarian Puskesmas Berbasis Android”. Pada penelitian ini masalah yang dihadapi yaitu Masyarakat di Indonesia menjadikan smartphone sebagai peralatan telekomunikasi utama, termasuk untuk mengakses internet. Tetapi, hal ini tidak diimbangi oleh pemanfaatan smartphone terhadap kesehatan. Buktinya, 4,6 juta anak balita di Indonesia

mengalami gizi buruk dan dapat bertambah setiap harinya. Kesadaran masyarakat masih sangat kurang mengenai kesehatan, terutama kesehatan balita dan ibu hamil. Maka dibuatkan aplikasi pemantau gizi balita dan ibu hamil serta pencarian puskesmas dengan menggunakan metode sequential search. hasil penelitian yang dilakukan berupa pengecekan waktu komputasi pencarian data, data usage, rata-rata RAM dan kesesuaian program pada Android itu sendiri. Aplikasi ini dapat membantu masyarakat dalam pengecekan secara berkala status gizi balita dan ibu hamil, rata-rata data usage 2.03 MB, rata-rata RAM 11.85 MB, dan waktu komputasi yang dibutuhkan dalam pencarian data adalah 44.13 ms untuk puskesmas, 0.013 ms untuk gizi balita, dan 0.011 ms untuk gizi ibu hamil.

4. Nurwanto dalam penelitian 2021 berjudul “Aplikasi Perpustakaan Berbasis Web dengan Mengimplementasikan Metode Sequential Search”. Permasalahan yang dihadapi pada penelitian ini yaitu pengunjung perpustakaan terus berkurang setiap harinya karena masyarakat lebih memilih membaca Electronic Book (E-Book) yang bisa didapatkan dengan mudah melalui internet. Maka dibuatkan aplikasi perpustakaan berbasis web dengan bahasa pemrograman PHP yang dapat digunakan untuk membantu pencarian buku dengan mengimplementasikan metode Sequential Search. Dengan demikian didapatkan hasil penelitian Sequential Search akurat yang dapat menampilkan data buku sesuai kata kunci yang dimasukan. Aplikasi akan mencari sesuai urutan indeks array yang telah

ditentukan yaitu kode buku, ISBN, judul buku, penulis dan penerbit. Hasil pengujian menggunakan metode Blackbox Testing dengan cara aplikasi di uji langsung oleh pengguna. aplikasi perpustakaan layak untuk diimplementasikan karena dari 9 skenario yang diujikan semuanya sukses dan sesuai hasil yang diharapkan.

5. Syaiful Bahri dalam penelitian 2019 berjudul “Implementasi Speech Recognition pada Aplikasi Belajar bentuk Permainan Menebak Kata Baku Bahasa Indonesia”. Pada penelitian ini Permasalahan yang sering dihadapi yaitu Pemilik android sering berhadapan dengan berbagai game atau permainan yang memiliki berbagai kategori, game yang sangat membantu dalam edukasi belajar kata baku adalah game yang berkategori pembelajaran. Sebuah aplikasi android dalam bentuk game atau permainan sebenarnya sangat penting untuk menarik para pengguna mainkan khususnya bagi anak-anak. Mata Pelajaran juga dapat di aplikasikan menjadi sebuah game yang menarik dan interaktif. Maka dibuatkan aplikasi menebak kata baku dengan metode speech recognition dengan hasil penelitian informasi yang disediakan oleh aplikasi ini mudah dimengerti dengan hasil Sangat setuju 30% setuju 70%, penggunaan menu atau fitur aplikasi menu mudah digunakan Sangat setuju 40% Setuju 60%, aplikasi nyaman digunakan Sangat setuju 50%, Setuju 50%.

2.2 Website

Menurut Kinaswara (2019:72) *Website* adalah suatu halaman web yang saling berhubungan yang umumnya berada pada peladen yang sama berisikan

kumpulan informasi yang disediakan secara perorangan, kelompok, atau organisasi. Sebuah situs web biasanya ditempatkan setidaknya pada sebuah server web yang dapat diakses melalui jaringan internet. “*Website* atau disingkat web, dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital baik berupa text, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur internet.

Lebih jelasnya, *website* merupakan halaman-halaman yang berisi informasi yang ditampilkan oleh browser seperti Mozilla Firefox, Google Chrome atau yang lainnya. Website merupakan fasilitas internet yang menghubungkan dokumen dalam lingkup lokal maupun jarak jauh. Dokumen pada website disebut dengan web page dan link dalam website memungkinkan pengguna bisa berpindah dari satu halaman ke halaman yang lain, baik diantara halaman yang disimpan dalam server maupun server diseluruh dunia.

2.3 Aplikasi

Menurut Voutama (2021:106) Aplikasi (juga disebut penerapan) adalah subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tetapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna. Contoh aplikasi adalah: pengolah kata, lembar kerja, dan pemutar media.

Beberapa aplikasi yang digabung bersama menjadi suatu paket terkadang disebut sebagai suatu "paket". Contohnya adalah Microsoft Office dan Open Office, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja, serta beberapa aplikasi lainnya. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi. Sering kali, mereka memiliki kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna. Contohnya, suatu lembar kerja dapat dibenamkan dalam suatu dokumen pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah.

2.4 PHP: *Hypertext Preprocessor*

Menurut Novendri (2019:46) PHP merupakan suatu bahasa pemrograman yang hanya dapat berjalan di web server. Seorang programmer Unix dan Perl bernama Rasmus Lerdoft merupakan pencipta dari PHP. Awalnya PHP hanya suatu kumpulan script biasa dan dengan bertambahnya waktu di berikan fitur pemrograman berorientasi objek (Subagia, 2018:1). PHP (Hypertext Preprocessor) merupakan sebuah script open source yang digunakan untuk mengembangkan sebuah website dan PHP dapat digabungkan ke dalam HTML. PHP (Hypertext Preprocessor) mengeksekusi setiap kodenya dilakukan di dalam server. Dengan cara seperti ini maka client tidak bias mengetahui pemrograman yang akan dibuat. Berdasarkan uraian di atas maka dapat disimpulkan bahwa PHP adalah suatu model bahasa pemrograman yang dapat digunakan dalam pembuatan website. PHP diciptakan oleh seorang programmer bernama Rasmus Lerdoft untuk membuat website pribadinya.

2.5 Javascript

Menurut Kamil (2023:3516) JavaScript adalah bahasa pemrograman web yang bersifat Client-Side Programming Language. Client-Side Programming Language adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh client. Javascript merupakan bahasa “case sensitive” yaitu membedakan perubahan variabel dan fungsi yang menggunakan huruf kecil dan besar.

Fungsi JavaScript cukup banyak, mulai dari membuat tampilan website jadi lebih menarik, menciptakan aplikasi mobile, mengembangkan game, hingga menjalankan web server. Fungsi JavaScript selanjutnya ialah untuk membuat aplikasi mobile. Kalau dulu aplikasi mobile hanya bisa dikembangkan dengan bahasa pemrograman Java untuk Android. Sekarang, bahasa pemrograman ini sudah bisa digunakan dalam mobile API untuk membuat aplikasi mobile.

2.6 Database

Menurut Ramadhan (2020:130) *Database* atau basis data adalah kumpulan data yang dikelola sedemikian rupa berdasarkan ketentuan tertentu yang saling berhubungan sehingga mudah dalam pengelolaannya. Melalui pengelolaan tersebut pengguna dapat memperoleh kemudahan dalam mencari informasi, menyimpan informasi dan membuang informasi.

Manfaat Database yaitu sebagai berikut:

a. Cepat dan Mudah

Database memberikan kemudahan dan kecepatan dalam menyeleksi data dan informasi. Namun, perlu digaris bawahi kalau setiap database punya kemampuan dan kecepatan yang berbeda.

b. Multi-User

Dalam waktu bersamaan, database bisa digunakan oleh beberapa pengguna sekaligus. Sementara penyimpanan terpusat tersimpan di satu unit.

c. Keamanan

Data yang tersimpan di dalam database hanya bisa diakses oleh pihak-pihak yang memiliki izin. Hampir semua database sudah menerapkan instrumen *password* sehingga keamanannya lebih terjaga.

d. Hemat Biaya

Database hanya memerlukan satu unit utama untuk menyimpan seluruh data. Perusahaan tidak memerlukan beberapa unit berbeda untuk penyimpanan data sehingga bisa lebih menghemat biaya.

e. Kontrol Data Terpusat

Pengaturan atau manajemen data dapat dilakukan oleh satu divisi sementara divisi lain bertugas mengumpulkan data di satu server.

f. Memudahkan Pembuatan Aplikasi

Proses pembuatan aplikasi menjadi lebih mudah karena programmer tidak perlu membuat struktur database baru. Cukup menggunakan struktur database lama untuk membuat aplikasi CRUD baru.

2.7 MySQL

Menurut Noviana (2022:114) MySQL adalah Sebuah program *database server* yang mampu menerima dan mengirimkan datanya sangat cepat, multi user serta menggunakan perintah dasar *Structured Query Language (SQL)*. MySQL merupakan dua bentuk lisensi, yaitu *FreeSoftware* dan *Shareware*. MySQL yang biasa kita gunakan adalah MySQL *FreeSoftware* yang berada dibawah *Lisensi General Public License*.

MySQL Merupakan sebuah *database server* yang *free*, artinya kita bebas menggunakan database ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya. MySQL pertama kali dirintis oleh seorang programmer database bernama Michael Widenius . Selain *database server*, MySQL juga merupakan program yang dapat mengakses suatu *database* MySQL yang berposisi sebagai *Server*, yang berarti program kita berposisi sebagai *Client*.

Fungsi MySQL adalah sebuah database yang dapat digunakan sebagai Client maupun server. Database MySQL merupakan suatu perangkat lunak database yang berbentuk database relasional atau disebut *Relational Database Management System (RDBMS)* yang menggunakan suatu bahasa permintaan yang bernama *Structured Query Language (SQL)*. Tipe data yang terdapat dalam *My Structured Query Language (MySQL)* secara umum sebagai berikut:

1. *Integer*

Ini menyimpan seluruh angka, baik positif atau pun negatif. Contohnya: 2, -45, 123 dan 23996. Pada contoh kita di atas, umur menggunakan data integer.

2. *Float*

Ini digunakan untuk menyimpan angka jika Anda ingin menggunakan desimal (koma). Contohnya: 2.5, -.455, 43.777 dan seterusnya.

3. *Datetime*

Untuk menyimpan data tanggal dan waktu dengan format: YYYY-MM-DD HH:MM:SS.

4. *Varchar*

Untuk menyimpan data teks terbatas atau pun karakter tunggal. Pada contoh kita, kolom nama menggunakan tipe data *varchar*.

5. *Blob*

Untuk menyimpan data *biner*, alih-alih data teks.

2.8 **Metode *Sequential Search***

Menurut Widodo (2021:86) *Sequential Search* adalah proses membandingkan setiap elemen *array* satu persatu secara beruntun dimulai dari elemen pertama hingga elemen yang dicari ditemukan atau hingga elemen terakhir dari *array*. Metode *Sequential Search* atau disebut pencarian beruntun dapat digunakan untuk melakukan pencarian data baik pada *array* yang sudah terurut maupun yang belum terurut. Proses yang terjadi adalah Membaca *array* data. Menentukan data yang dicari. Mulai dari data pertama sampai dengan data terakhir, data yang dicari dibandingkan dengan masing-masing data di dalam *array*. Jika data yang dicari tidak ditemukan maka semua data atau elemen *array* dibandingkan

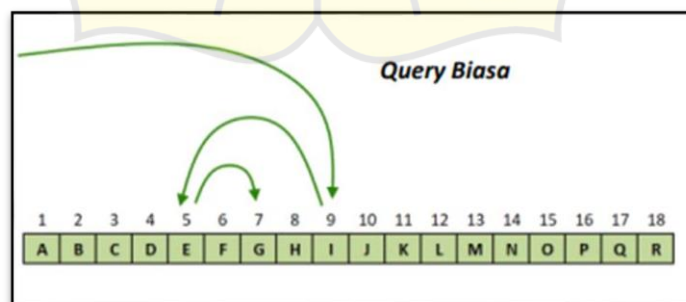
sampai selesai. Jika data yang dicari ditemukan maka perbandingan akan dihentikan. Proses pencarian data dengan metode ini cukup sederhana dan mudah.

Proses pencarian data dilakukan dengan mencocokkan data yang dilakukan secara berurut satu demi satu dimulai dari data ke-1 hingga data pada urutan terakhir. Jika data yang dicari mempunyai nilai yang sama dengan data yang ada dalam kelompok data, berarti data telah ditemukan. Jika data yang dicari tidak ada yang cocok dalam sekelompok data, data tersebut tidak ada dalam sekelompok data. Selanjutnya kita tinggal menampilkan hasil yang diperoleh tersebut.

Kelebihan Metode *Sequential Search*:

1. Pencarian dapat digunakan untuk pemecahan masalah (*wide applicability*).
2. Pencarian dapat dilakukan secara langsung dan jelas dan apabila data yang dicari ada pada elemen pertama maka hanya perlu dilakukan satu kali proses dan data segera ditemukan.
3. Dapat digunakan pada data yang tidak terurut.

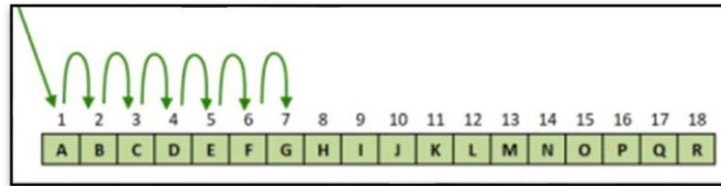
Perbandingan *Query Biasa* dengan *Sequential Search*:



Gambar 2.1 Proses *Query Biasa*

Berdasarkan gambar 2.1 kita dapat melihat proses pada *Query biasa* dimana proses

jalannya secara acak sesuai pada tanda panah untuk dapat menampilkan hasil.



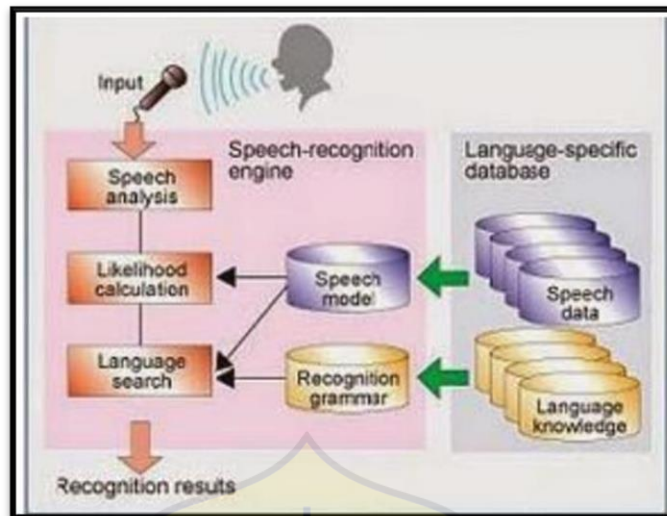
Gambar 2.2 Proses *Sequential Search*

Berdasarkan gambar 2.2 di atas kita dapat mengetahui bagaimana proses dari pada *Sequential Search* dimana anak panahnya berjalan secara berurutan untuk dapat menampilkan hasil

Berdasarkan gambar 2.1 dan gambar 2.2 tersebut dapat disimpulkan bahwa jika menggunakan *query* biasa maka data yang dicari tidak berurut sedangkan pada metode *Sequential Search* pencarian dilakukan berurut dengan mengecek data satu persatu sehingga hasilnya lebih optimal

2.9 Metode *Automatic Speech Recognition*

Menurut Farizi (2022:158) *Automatic Speech recognition* merupakan proses pengidentifikasian yang dilakukan oleh komputer ataupun perangkat smartphone untuk mengenali kata yang diucapkan oleh manusia tanpa memperdulikan identitas pemberi suara. Proses identifikasi kata dilakukan dengan mengkonversi sinyal akustik yang ditangkap oleh audio device. Pola kerja pengenalan ujaran (*speech recognition*) yaitu dengan mencocokkan sinyal akustik yang diterima dengan data yang tersimpan dalam template atau database.



Gambar 2.3 Skema *Speech Recognition*

Speech recognition merupakan sistem yang digunakan untuk dapat mengenali perintah kata dari suara manusia dan kemudian diterjemahkan menjadi suatu data yang dapat dimengerti oleh komputer. Pada saat ini, sistem *Speech Recognition* ini banyak digunakan untuk menggantikan peranan input dari keyboard dan mouse. Dengan penggunaan sistem *Speech Recognition* ini terdapat keuntungan berupa kecepatan dan kemudahan dalam penggunaannya.

Perangkat pengenal ucapan (*speech recognition*) memiliki 3 tahapan dalam prosesnya, yaitu:

1. Penerimaan Masukan:

Masukan atau input berupa kata-kata atau perintah yang diucapkan oleh manusia melalui *microfon*.

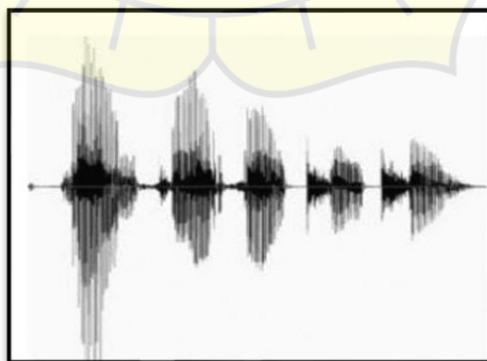
2. Ekstraksi:

Tahap ini merupakan tahapan penyimpanan masukan yang berupa input suara sekaligus untuk pembuatan basis data sebagai pola. Pada tahap ini dilakukan

pengaturan dari hasil suara input berupa pengaturan amplitudo, frekuensi, dan fase dari gelombang sinus untuk mendapatkan data yang akan dibandingkan dengan *database*.

3. Tahap Perbandingan:

Pada tahap ini merupakan tahapan pencocokan data baru dengan data suara (pencocokan tata bahasa) pada pola. Tahapan ini dimulai dengan mengkonversi sinyal suara digital hasil dari proses ekstraksi menjadi bentuk spektrum suara yang akan dianalisa dengan membandingkan dengan pola suara pada basis data. Sebelumnya, data-data suara masukan dipisah-pisah dan kemudian diproses satu per satu berdasarkan urutannya. Pemilihan ini dilakukan dengan tujuan agar proses analisis data yang didapatkan dilakukan secara paralel. Secara umum, speech recognizer memproses sinyal suara yang masuk dan menyimpannya dalam bentuk digital. Hasil proses digitalisasi tersebut kemudian dikonversi dalam bentuk spektrum suara yang akan dianalisa dengan membandingkannya dengan template suara pada database sistem.



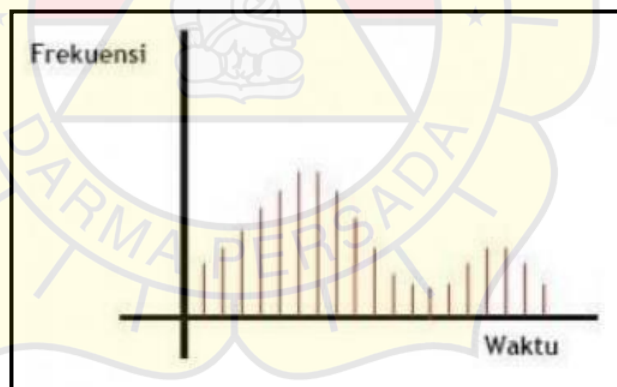
Gambar 2.4 Spektrum Suara

Sebelumnya, data suara masukan dipilah-pilah dan diproses satu per satu berdasarkan urutannya. Pemilahan ini dilakukan agar proses analisis dapat

dilakukan secara paralel. Proses yang pertama kali dilakukan ialah memproses gelombang kontinu spektrum suara ke dalam bentuk diskrit. Langkah berikutnya ialah proses kalkulasi yang dibagi menjadi dua bagian:

- a. Transformasi gelombang diskrit menjadi *array* data.
- b. Untuk masing-masing elemen pada *array* data, dilakukan perhitungan gelombang (frekuensi).

Hasil masukan dari sinyal akustik dirubah dengan ADC (*Analog to Digital Converter*) menjadi data diskrit gelombang suara. Ketika mengkonversi gelombang suara ke dalam bentuk diskrit, gelombang diperlebar dengan cara memperinci berdasarkan waktu. Hal ini dilakukan agar proses algoritma selanjutnya atau pencocokan lebih mudah dilakukan.



Gambar 2.5 Hasil Konversi Sinyal Diskrit

Dari tiap elemen array data tersebut, dikonversi ke dalam bentuk bilangan biner.

Data biner tersebut yang nantinya akan dibandingkan dengan template data suara

Berikut Detail *Speech Recognition* yang dirancang:

1. Menggunakan *Library Javascript*

Menggunakan Library dengan bahasa penyusun Javascript yaitu Web Speech API yang mendukung browsers dan memiliki fitur memasukkan data suara ke dalam aplikasi web. Web Speech API memiliki dua bagian: *Speech Synthesis (Text-to-Speech)*, dan *Speech Recognition (Asynchronous Speech Recognition)*.

2. Perangkat Pendukung

1. *Default Microphone*

Dapat menggunakan audio bawaan dari perangkat sistem operasi yang umumnya sudah tersedia.

2. *External Microphone*

Dapat menggunakan perangkat tambahan berupa eksternal microphone sebagai input.

3. Browsers

Dapat menggunakan perambanan jelajah atau browser yang sudah mendukung fitur akses audio diantara Google Chrome dan Mozilla Firefox

3. Penggunaan *Artificial Intelligence*

Menurut Lubis (2021:1) menggunakan Algoritma *Natural language Processing*. *Natural Language Processing* atau biasa disingkat NLP, adalah salah satu cabang *artificial intelligence* atau kecerdasan buatan yang berhubungan dengan interaksi antara komputer dan manusia menggunakan bahasa alami. Tujuannya adalah memberikan kemampuan kepada mesin

agar dapat membaca, memahami, dan memperoleh makna dari bahasa manusia. Selain itu, *speech-to-text* pada *speech recognition* juga memudahkan kita untuk menuliskan apa yang ingin kita sampaikan di gawai (gadget) tanpa harus mengetik. Oleh karena itu, teknologi ini juga hadir di berbagai aplikasi. Contoh *natural language processing speech to text* adalah aplikasi Microsoft Office hingga aplikasi asisten pribadi seperti Siri, Cortana, dan Google Assistant.

2.10 Unified Modeling Language (UML)

Menurut A.S. Rosa (2018:30) *Unified Modeling Language* terbagi menjadi sembilan diagram yang masing-masing memiliki aturan-aturan tertentu dalam penyusunan. Diagram - diagram tersebut tersusun atas sejumlah elemen grafik yang saling membentuk satu kesatuan dalam pemodelan *software*. Masing-masing diagram UML mempresentasikan berbagai sudut pandangan terhadap sistem dan mendefinisikan apa yang dikerjakan oleh sistem, bukan bagaimana sistem berkerja. Kesembilan diagram UML tersebut dibagi menjadi tiga kelompok berdasarkan fungsinya, yaitu:

1. Diagram untuk requirement dan desain, terdiri dari tujuh diagram, di antaranya adalah *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, *Object Diagram*, *Sequence Diagram*, *Collaboration Diagram*, *State Diagram*.
2. Diagram mengenai organisasi umum *software*, terdiri dari satu diagram, yaitu *Diagram Package*



3. Diagram untuk implementasi, terdiri dari suatu diagram, yaitu *Component & Deployment Diagram*.

Merupakan diagram yang dibuat pada awal pemodelan suatu *software*, karena diagram ini memberikan penjelasan umum antara sistem dengan “dunia luar” serta fitur – fitur apa yang harus dimiliki oleh sistem jika dipandang dari dunia luar tersebut, *Use case* diagram menjelaskan manfaat dari aplikasi jika dilihat dari sudut pandang orang yang berada diluar sistem (aktor). Diagram ini menunjukkan fungsi analitik suatu sistem atau kelas dan bagaimana sistem berinteraksi dengan dunia luar.

2.10.1 Use Case Diagram

Use Case diagram dapat digunakan selama proses analisa untuk menangkap *requirement* atau permintaan terhadap sistem dan untuk memahami bagaimana sistem tersebut harus bekerja: Macam-macam simbol *Use Case Diagram* adalah seperti pada tabel 2.1 berikut ini:

Tabel 2.1 Simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1.		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).

No	Gambar	Nama	Keterangan
3.		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4.		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5.		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7.		<i>Sistem</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas
8.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
9.		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10.		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber: Rosa.AS, M. Shalahuddin, 2018

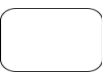





2.10.2 Activity Diagram

Activity diagram adalah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja. *Activity diagram* mempunyai peran seperti halnya

flowchart, tetapi perbedaannya adalah *activity* diagram bisa mendukung perilaku paralel sementara *flowchart* tidak bisa. Diagram ini bersifat dinamis suatu aktifitas lainnya dalam suatu sistem.

Macam-macam simbol *Activity* Diagram adalah seperti pada tabel 2.2 berikut ini:

Tabel 2.2 Simbol *Activity* Diagram

No	Gambar	Nama	Keterangan
1.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2.		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3.		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4.		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5.		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran
6.		<i>Join Node</i>	Join node atau node gabung adalah node kontrol yang menyinkronkan banyak aliran. Join node memiliki beberapa tepi masuk dan satu tepi keluar.

Sumber: Rosa.AS, M. Shalahuddin, 2018

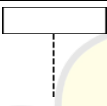


2.10.3 Sequence Diagram

Sequence diagram menjelaskan interaksi objek yang disusun dalam suatu urutan waktu, Diagram ini secara khusus berasosiasi dengan use-case. Sequence diagram memperlihatkan tahap demi tahap apa yang harus terjadi untuk

menghasilkan suatu didalam *use-case* diagram. Tipe diagram yang digunakan sebaiknya digunakan diawal tahap desain atau analisis karena kesederhanaannya dan mudah untuk di mengerti.

Macam-macam simbol *Sequence* Diagram adalah seperti pada tabel 2.3 berikut ini:

Tabel 2.3 Simbol *Sequence* Diagram

No	Gambar	Nama	Keterangan
1.		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Sumber: Rosa.AS, M. Shalahuddin, 2018





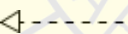
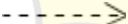

2.10.4 Class Diagram

Class diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem/perangkat lunak yang sedang kita kembangkan. *Class* diagram memberi kita gambaran/diagram statis tentang sistem/perangkat lunak dan relasi-relasi yang ada di dalamnya.

Dalam kelas. Seperti objek, adalah sesuatu yang membungkus (encapsulate) informasi (baca: atribut) dan perilaku (baca: operasi) dalam dirinya. Dalam pengembangan sistem tradisional, kita mengadakan pendekatan dengan cara

memisahkan informasi-informasi pada sisi basis data dan perilaku yang mengaksesnya di sisi aplikasi pengakses. Macam-macam simbol *Class Diagram* adalah seperti pada tabel 2.4 berikut ini:

Tabel 2.4 Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Sumber: Rosa.AS, M. Shalahuddin, 2018