

SKRIPSI

**ASISTEN PEMBERI REKOMENDASI MENU FAVORIT YANG
INTERAKTIF DENGAN SUARA**



Josie Esthaliani

NPM: 6181801008

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2023**

UNDERGRADUATE THESIS

**FAVORITE MENU RECOMMENDER ASSISTANT
INTERACTING WITH VOICE**



Josie Esthaliani

NPM: 6181801008

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2023**

LEMBAR PENGESAHAN

ASISTEN PEMBERI REKOMENDASI MENU FAVORIT YANG INTERAKTIF DENGAN SUARA

Josie Esthaliani

NPM: 6181801008

Bandung, 18 Januari 2023

Menyetujui,

Pembimbing

Digitally signed
by Veronica Sri
Moertini

Dr. Veronica Sri Moertini

Ketua Tim Penguji

Digitally signed
by Natalia

Natalia, M.Si.

Anggota Tim Penguji

Digitally signed
by Keenan
Adiwijaya Leman

Keenan Adiwijaya Leman, M.T.

Mengetahui,

Ketua Program Studi

Digitally signed
by Mariskha Tri
Adithia

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

ASISTEN PEMBERI REKOMENDASI MENU FAVORIT YANG INTERAKTIF DENGAN SUARA

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 18 Januari 2023



Josie Esthaliani
NPM: 6181801008

ABSTRAK

Makanan merupakan kebutuhan primer manusia yang pada umumnya dihasilkan melalui proses yang disebut memasak. Semakin maju dan berkembangnya dunia memasak menghasilkan beragam variasi menu masakan. Beragamnya menu dapat membuat seseorang bingung dalam memilih menu dan berakhir dengan menu yang secara repetitif dipilih dan dapat menimbulkan rasa bosan. Agar tidak bosan perlu ada jeda dari pemilihan menu masakan yang sama. Informasi mengenai pembuatan menu masakan dapat diperoleh dari resep masakan. Dengan jangka waktu pemilihan menu agar tidak repetitif dan informasi yang ada pada resep dapat menjadi penentu pemilihan menu masakan. Dengan banyaknya pertimbangan dalam menentukan pemilihan menu dapat membuat seseorang semakin bingung dalam memilih menu masakan. Rekomendasi dapat diberikan oleh sistem dengan mencocokkan kebutuhan pengguna dengan informasi yang dimiliki sistem. Oleh karena itu kebutuhan pengguna juga menjadi penentu dalam memilih menu sehingga perangkat lunak perlu dapat menerima informasi tersebut. *Chatbot* memungkinkan perangkat menerima informasi langsung dari pengguna maupun memberikan informasi pada pengguna. *Chatbot* didesain untuk melakukan percakapan dengan mengikuti karakteristik komunikasi manusia. *Text-to-speech* dan *speech-to-text* dapat membuat proses interaksi menjadi interaktif dengan suara. *Text-to-speech* dapat mengubah teks menjadi suara dan *speech-to-text* dapat mengubah suara menjadi teks. Google Cloud Platform(GCP) menyediakan API untuk dapat melakukan *text-to-speech* dan *speech-to-text*.

Pada penelitian ini, dibangun sebuah perangkat lunak yang dapat memberikan rekomendasi menu favorit yang sudah disesuaikan dengan *input* yang diterima secara interaktif dengan suara. Perangkat lunak ini dapat menjadi solusi dalam memilih menu makanan dan sesuai dengan kebutuhan pengguna. Perangkat lunak mengimplementasikan *text-to-speech* dan *speech-to-text* pada *chatbot* untuk menerima dan memberikan informasi pada pengguna dengan suara. *Chatbot* yang diimplementasikan merupakan *rule-based chatbot* dengan domain tertutup. Pengguna akan menerima pertanyaan yang terus mengarahkan pada jawaban hasil rekomendasi dengan menerima *input* dari karakteristik makanan yang diinginkan. Hasil rekomendasi diberikan dengan melakukan pencocokan kebutuhan pengguna dengan data dengan menggunakan *cosine similarity*. Hasil diberikan dengan mempertimbangkan jangka waktu tunggu agar menu tidak repetitif dan skor kecocokan permintaan dari *input* dengan informasi menu masakan.

Berdasarkan hasil pengujian perangkat lunak dapat memberikan hasil rekomendasi menu favorit yang sesuai dengan kebutuhan. *Chatbot* sudah dapat berjalan sesuai dengan alur yang dibangun untuk setiap kasus percakapan. Seluruh fitur pada perangkat lunak sudah dapat berjalan dengan baik, walau terdapat beberapa kata yang belum dapat diterjemahkan oleh *speech-to-text* dari API GCP dengan baik. Walau begitu perangkat lunak tetap dapat menerima *input* dalam bentuk teks sehingga menjadi solusi jika ada kata yang tidak dapat diterjemahkan dengan baik.

Kata-kata kunci: Rekomendasi, Chatbot, Rule-Based, Cosine Similarity, Menu Masakan, speech to text, text to speech

ABSTRACT

Food is a primary human need which is generally produced through a process called cooking. The more advanced and developing world makes a variety of food menu variations. The variety of menus can make someone confused in choosing a menu and end up with the menu being selected repeatedly. Information about food preparation can be obtained from cooking recipes. With a menu selection period so that it is not repeated and the information contained in the recipe can be a determinant of choosing a food menu. With so many considerations in determining menu selection, it can make someone even more confused in choosing a food menu. Recommendations can be provided by the system by matching user needs with information owned by the system. Therefore the user's needs are also a determinant in choosing a menu so that the software needs to be able to receive this information. Chatbot allows devices to receive information directly from users and provide information to users. Chatbots are designed to carry out conversations by following human communication features. Text-to-speech and speech-to-text can make the interaction process interactive with voice. Text-to-speech can convert text to voice and speech-to-text can convert voice to text. Google Cloud Platform provides an API to be able to do text-to-speech and speech-to-text.

In this study, a software was built that can provide recommendations for favorite menus that have been adjusted to the input received interactively with sound. This software can be a solution in choosing a food menu and according to user needs. The software implements text-to-speech and speech-to-text on chatbots to receive and provide information to users. The implemented chatbot is a rule-based chatbot with a closed domain. Users will receive questions that continue to lead to answers to recommendations by receiving input from the desired food characteristics. The recommendation results are given by matching user needs with data using cosine similarity. The results are given considering the waiting period so that the menu does not repeat itself and the demand matches the score of the input with the food.

Based on the results of testing the software can provide recommendations for favorite menus that suit your needs. All features in the software are running well, although there are some words that GCP's speech-to-text cannot be translated properly. Even so, the software can still accept input in text form so that it becomes a solution if there are words that cannot be translated properly.

Keywords: Recommendation, Chatbot, Rule-Based, Cosine Similarity, Food, speech to text, text to speech

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan yang Maha Esa karena atas kasih dan karuniaNya penulis dapat menyelesaikan penyusunan skripsi yang berjudul "Asisten Pemberi Rekomendasi Menu Favorit Yang Interaktif Dengan Suara". Penulis menyadari bahwa dalam pengerjaan skripsi ini penulis tidak luput dari kesalahan dan kendala. Akan tetapi, banyak pihak yang telah membantu membimbing, memberikan dukungan dan motivasi untuk menyelesaikan skripsi ini. Sehingga penulis menyadari pengerjaan skripsi ini tidak mungkin dilakukan tanpa dukungan dan bantuan orang-orang terdekat. Oleh karena itu, penulis ingin menyampaikan secara khusus rasa terima kasih kepada:

- Keluarga penulis terutama orang tua dan adik yang telah memberikan dukungan, semangat dan doa selama pengerjaan skripsi ini.
- Ibu Dr. Veronica Sri Moertini, Ir., M.T. selaku dosen pembimbing yang telah meluangkan waktunya untuk membimbing, mendampingi dan memberikan arah kepada penulis sehingga skripsi ini dapat selesai.
- Grace Arlenne Suganda selaku teman yang memberikan dukungan dan membantu penulis memahami informasi dari resep masakan.
- Teman-teman semasa kuliah, khususnya Kezia, Afi, Juan, Lukas, Wira, Carizza, dan Adira yang telah menemani selama masa perkuliahan dan memberikan dukungan selama pengerjaan skripsi ini.
- Rama dan teman-teman satu dosen pembimbing yang senantiasa saling menyemangati dan mendukung selama pengerjaan skripsi ini.
- Teman-teman dan sahabat penulis yang tidak dapat disebut satu-persatu yang terus memberikan semangat, dukungan, dan mendengarkan keluh kesah penulis selama pengerjaan skripsi ini.
- Pihak-pihak lain yang telah membantu penulis baik secara langsung maupun tidak langsung sehingga penulis dapat menyelesaikan skripsi ini.

Penulis menyadari bahwa penelitian ini masih jauh dari kata sempurna. Maka dari itu, penulis terbuka akan kritik dan saran yang dapat membangun dan menyempurnakan penelitian ini. Semoga penelitian ini dapat bermanfaat bagi pembaca dan menginspirasi untuk penelitian-penelitian selanjutnya.

Bandung, Januari 2023

Penulis

DAFTAR ISI

| | |
|---|-----------|
| KATA PENGANTAR | xv |
| DAFTAR ISI | xvii |
| DAFTAR GAMBAR | xix |
| DAFTAR TABEL | xxi |
| DAFTAR KODE PROGRAM | xxiv |
| 1 PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 2 |
| 1.3 Tujuan | 2 |
| 1.4 Batasan Masalah | 2 |
| 1.5 Metodologi | 2 |
| 1.6 Sistematika Pembahasan | 3 |
| 2 LANDASAN TEORI | 5 |
| 2.1 <i>Data Science</i> [1] | 5 |
| 2.2 <i>Chatbot</i> | 7 |
| 2.3 Natural Language Processing | 11 |
| 2.3.1 Regular Expression | 11 |
| 2.3.2 Tokenisasi | 12 |
| 2.3.3 Lematisasi | 13 |
| 2.3.4 Stemming | 13 |
| 2.3.5 <i>Part-of-speech tagging</i> | 14 |
| 2.4 <i>Vector Space Model</i> | 14 |
| 2.5 <i>Cosine Similarity</i> | 15 |
| 2.6 <i>Speech-to-Text</i> [2] | 16 |
| 2.7 <i>Text-to-Speech</i> [3] | 17 |
| 2.8 <i>Web Scraping</i> [4] | 18 |
| 2.8.1 <i>Library BeautifulSoup</i> | 18 |
| 2.8.2 <i>Library Selenium</i> | 20 |
| 2.9 Google Cloud Platform | 20 |
| 2.9.1 Google Cloud Speech API | 21 |
| 2.9.2 Google Cloud Text-to-Speech API | 21 |
| 3 EKSPLORASI TEKNOLOGI | 23 |
| 3.1 Eksplorasi Penarikan Data dengan Teknik <i>Web Scraping</i> | 23 |
| 3.1.1 <i>Library Selenium</i> | 23 |
| 3.1.2 <i>Library BeautifulSoup</i> | 25 |
| 3.2 Eksplorasi Google Cloud Platform | 27 |

| | | |
|----------|---|-----------|
| 3.2.1 | Speech-to-Text API | 27 |
| 3.2.2 | Text-to-Speech API | 28 |
| 4 | ANALISIS DAN RANCANGAN | 29 |
| 4.1 | Analisis Masalah | 29 |
| 4.2 | Analisis Kebutuhan | 29 |
| 4.3 | Ide Solusi | 31 |
| 4.4 | Analisis Chatbot | 34 |
| 4.5 | Pengumpulan dan penyiapan data | 35 |
| 4.5.1 | Pengumpulan Data | 35 |
| 4.5.2 | Penyiapan Data | 37 |
| 4.6 | Analisis Arsitektur Perangkat Lunak | 42 |
| 4.7 | Rancangan Struktur Data | 45 |
| 4.8 | Rancangan Algoritma | 47 |
| 5 | IMPLEMENTASI DAN PENGUJIAN PERANGKAT LUNAK | 51 |
| 5.1 | Implementasi | 51 |
| 5.1.1 | Implementasi Antarmuka | 51 |
| 5.1.2 | Implementasi <i>Rule Chatbot</i> | 52 |
| 5.1.3 | Implementasi Modul Program | 53 |
| 5.2 | Pengujian Perangkat Lunak | 60 |
| 5.2.1 | Kasus Pengujian | 60 |
| 5.2.2 | Hasil Pengujian | 64 |
| 5.2.3 | Kesimpulan Pengujian | 79 |
| 6 | KESIMPULAN DAN SARAN | 81 |
| 6.1 | Kesimpulan | 81 |
| 6.2 | Saran | 81 |
| | DAFTAR REFERENSI | 83 |
| | A KODE PROGRAM | 85 |
| | B DATA RESEP | 95 |

DAFTAR GAMBAR

| | | |
|------|---|----|
| 2.1 | Tahapan <i>data science</i> | 5 |
| 2.2 | Contoh ilustrasi tahapan eksplorasi dan penyiapan data | 6 |
| 2.3 | Contoh ilustrasi tahapan analisis data | 6 |
| 2.4 | Contoh alur percakapan | 8 |
| 2.5 | Ilustrasi <i>rule-based chatbot</i> | 8 |
| 2.6 | Ilustrasi <i>Retrieval-based chatbot</i> | 9 |
| 2.7 | Ilustrasi <i>Generation-based chatbot</i> | 10 |
| 2.8 | Struktur AIML | 11 |
| 2.9 | Contoh pos tag dari kata dan hasil pos tag | 14 |
| 2.10 | Arsitektur proses <i>speech to text</i> | 16 |
| 2.11 | Alur proses <i>Text to Speech</i> | 17 |
| 2.12 | Struktur objek BeautifulSoup | 19 |
| | | |
| 3.1 | Halaman <i>website</i> saat sampai pada bagian dasar halaman | 24 |
| 3.2 | Halaman <i>website</i> pada bagian akhir dasar halaman setelah penerapan Selenium | 25 |
| 3.3 | Contoh hasil perbedaan output dari <i>find</i> dan <i>findAll</i> | 26 |
| 3.4 | Hasil <i>web scraping</i> pada halaman kumpulan resep | 26 |
| 3.5 | Hasil <i>web scraping</i> dengan informasi lengkap | 27 |
| 3.6 | Hasil <i>Speech-to-Text</i> | 27 |
| 3.7 | Hasil <i>Speech-to-Text</i> dengan hasil kurang baik | 28 |
| | | |
| 4.1 | Grafik Jawaban pertanyaan 1 | 30 |
| 4.2 | Grafik Jawaban pertanyaan 2 | 30 |
| 4.3 | Grafik Jawaban pertanyaan 3 | 31 |
| 4.4 | Alur percakapan dari <i>Chatbot</i> pemberi rekomendasi | 32 |
| 4.5 | Potongan data hasil <i>web scraping</i> tahap pertama | 36 |
| 4.6 | Potongan data hasil <i>web scraping</i> tahap kedua | 36 |
| 4.7 | Data Flow Diagram level 0 | 42 |
| 4.8 | Data Flow Diagram level 1 tahap 1 | 43 |
| 4.9 | Data Flow Diagram level 1 tahap 2 | 44 |
| 4.10 | ERD dari data untuk Perangkat Lunak | 45 |
| 4.11 | <i>Flowchart</i> Rancangan <i>Chatbot</i> | 47 |
| 4.12 | Contoh penggunaan <i>rule</i> dengan <i>input</i> | 48 |
| | | |
| 5.1 | Tampilan <i>chatbot</i> hasil penggunaan | 51 |
| 5.2 | Tampilan <i>chatbot</i> | 52 |
| 5.3 | Flowchart method <i>get_response</i> | 54 |
| 5.4 | Flowchart method <i>check_all_messages</i> | 55 |
| 5.5 | Flowchart method <i>message_probability</i> | 57 |
| 5.6 | Flowchart method <i>filter_item</i> | 58 |
| 5.7 | Flowchart method <i>search menu</i> | 59 |
| 5.8 | Tangkapan layar kasus pengujian ke-1 | 65 |
| 5.9 | Tangkapan layar kasus pengujian ke-1 (lanjutan) | 65 |

| | | |
|------|--|----|
| 5.10 | Tangkapan layar kasus pengujian ke-2 | 66 |
| 5.11 | Tangkapan layar kasus pengujian ke-3 | 67 |
| 5.12 | Tangkapan layar kasus pengujian ke-4 | 68 |
| 5.13 | Tangkapan layar kasus pengujian ke-4 (lanjutan) | 68 |
| 5.14 | Tangkapan layar kasus pengujian ke-4 (lanjutan) | 69 |
| 5.15 | Tangkapan layar kasus pengujian ke-5 | 70 |
| 5.16 | Tangkapan layar kasus pengujian ke-5 (lanjutan) | 70 |
| 5.17 | Tangkapan layar kasus pengujian ke-6 | 71 |
| 5.18 | Tangkapan layar kasus pengujian ke-6 (lanjutan) | 72 |
| 5.19 | Tangkapan layar kasus pengujian ke-7 | 73 |
| 5.20 | Tangkapan layar kasus pengujian ke-7 (lanjutan) | 73 |
| 5.21 | Tangkapan layar kasus pengujian ke-8 | 74 |
| 5.22 | Tangkapan layar kasus pengujian ke-8 (lanjutan) | 75 |
| 5.23 | Tangkapan layar kasus pengujian ke-9 | 76 |
| 5.24 | Tangkapan layar kasus pengujian ke-9 (lanjutan) | 76 |
| 5.25 | Tangkapan layar kasus pengujian ke-10 | 77 |
| 5.26 | Tangkapan layar kasus pengujian ke-10 (lanjutan) | 78 |

DAFTAR TABEL

| | | |
|------|---|----|
| 2.1 | Aturan Regex[4] | 12 |
| 2.2 | Contoh aturan penulisan untuk <i>stemming</i> [5] | 13 |
| 2.3 | Contoh <i>Term-frequency vector</i> [6] | 15 |
| 4.1 | <i>Rule</i> untuk <i>chatbot</i> | 33 |
| 4.2 | Data akhir yang dibutuhkan perangkat lunak | 34 |
| 4.3 | <i>Rule</i> untuk <i>chatbot</i> | 34 |
| 4.4 | Data hasil <i>web scraping</i> halaman kumpulan resep | 35 |
| 4.5 | Data hasil <i>web scraping</i> setiap halaman resep | 36 |
| 4.6 | Contoh isi variabel nama menu, url, dan nama menu yang dibutuhkan | 37 |
| 4.7 | Contoh isi variabel bahan dan isi bahan yang seharusnya | 38 |
| 4.8 | Bahan dengan penulisan berbeda beserta penulisan yang benar | 40 |
| 4.9 | Pembagian kategori durasi memasak | 40 |
| 4.10 | Pembagian kategori dari tag | 40 |
| 4.11 | Tranformasi pada isi variabel tingkat kesulitan | 41 |
| 4.12 | Rancangan <i>rule chatbot</i> dari alur | 46 |
| 4.13 | Hasil contoh perhitungan vektor | 49 |
| 5.1 | <i>Input</i> untuk kasus pengujian ke-1 | 61 |
| 5.2 | <i>Input</i> untuk kasus pengujian ke-2 | 61 |
| 5.3 | <i>Input</i> untuk kasus pengujian ke-3 | 61 |
| 5.4 | <i>Input</i> untuk kasus pengujian ke-4 | 62 |
| 5.5 | <i>Input</i> untuk kasus pengujian ke-5 | 62 |
| 5.6 | <i>Input</i> untuk kasus pengujian ke-6 | 63 |
| 5.7 | <i>Input</i> untuk kasus pengujian ke-7 | 63 |
| 5.8 | <i>Input</i> untuk kasus pengujian ke-8 | 64 |
| 5.9 | <i>Input</i> untuk kasus pengujian ke-9 | 64 |
| 5.10 | <i>Input</i> untuk kasus pengujian ke-10 | 64 |
| 5.11 | Hasil skor menu rekomendasi kasus pengujian ke-1 | 65 |
| 5.12 | Hasil skor menu rekomendasi kasus pengujian ke-2 | 66 |
| 5.13 | Hasil skor menu rekomendasi kasus pengujian ke-3 | 67 |
| 5.14 | Hasil skor menu rekomendasi kasus pengujian ke-4 | 69 |
| 5.15 | Hasil skor menu rekomendasi kasus pengujian ke-5 | 70 |
| 5.16 | Hasil skor menu rekomendasi kasus pengujian ke-6 | 72 |
| 5.17 | Hasil skor menu rekomendasi kasus pengujian ke-7 | 73 |
| 5.18 | Hasil skor menu rekomendasi kasus pengujian ke-8 | 75 |
| 5.19 | Hasil skor menu rekomendasi kasus pengujian ke-9 | 76 |
| 5.20 | Hasil skor menu rekomendasi kasus pengujian ke-10 | 78 |
| B.1 | 5 data teratas resep | 95 |
| B.2 | 5 data terbawah resep | 99 |

DAFTAR KODE PROGRAM

| | | |
|------|--|----|
| 2.1 | Tokenisasi dengan NLTK [7] | 12 |
| 2.2 | Lemmatisasi dengan NLTK [7] | 13 |
| 2.3 | Stemming dengan NLTK[7] | 13 |
| 2.4 | Pos tag dengan NLTK [7] | 14 |
| 2.5 | Instalasi Library BeautifulSoup | 19 |
| 2.6 | Mendapatkan BeautifulSoup <i>object</i> dengan menjalankan <i>library</i> BeautifulSoup dan <i>request</i> | 19 |
| 2.7 | Penggunaan fungsi <i>find</i> dan <i>findAll</i> | 19 |
| 2.8 | Pendefinisian <i>driver</i> selenium | 20 |
| 2.9 | Penggunaan hasil <i>driver</i> Selenium untuk BeautifulSoup | 20 |
| 2.10 | Konfigurasi <i>speech recognition</i> | 21 |
| 2.11 | <i>request</i> Text-to-Speech API | 21 |
| | | |
| 3.1 | Penerapan Selenium untuk bergerak ke bagian dasar halaman | 24 |
| 3.2 | Penerapan Selenium untuk bergerak ke bagian dasar halaman | 24 |
| 3.3 | Mengambil halaman utama website | 25 |
| 3.4 | Penerapan <i>environment</i> | 27 |
| 3.5 | Instansiasi <i>Speech-to-Text</i> API | 27 |
| 3.6 | menentukan bahasa dan tipe <i>audio</i> | 27 |
| 3.7 | Melakukan permintaan <i>Speech-to-Text</i> | 27 |
| 3.8 | Instansiasi <i>Text-to-Speech</i> API | 28 |
| 3.9 | Menentukan bahasa dan tipe suara pada parameter <i>voiceId</i> | 28 |
| 3.10 | Menentukan bentuk file audio pada parameter <i>audioconfigId</i> | 28 |
| 3.11 | Mengisi parameter <i>synthesis input</i> | 28 |
| 3.12 | melakukan permintaan <i>Text-to-Speech</i> | 28 |
| | | |
| 4.1 | Penyiapan variabel nama menu | 37 |
| 4.2 | Membuang keterangan bahan | 38 |
| 4.3 | Pembersihan bahan dari keterangan dan <i>measurement</i> | 39 |
| 4.4 | Mengubah tingkat kesulitan sesuai standar yang sudah ditentukan | 41 |
| 4.5 | Menambah variabel jam makan | 41 |
| | | |
| 5.1 | <i>rule</i> JSON dari alur <i>chatbot</i> | 52 |
| 5.2 | Method <i>get_response</i> | 54 |
| 5.3 | Method <i>check_all_messages</i> | 56 |
| 5.4 | Method <i>message_probability</i> | 57 |
| 5.5 | Method <i>filter_item</i> | 58 |
| 5.6 | Method <i>searchMenu</i> | 60 |
| | | |
| A.1 | Modul Chatbot | 85 |
| A.2 | Modul Text-to-Speech | 88 |
| A.3 | Modul Speech-to-Text | 89 |
| A.4 | Modul Get Audio | 89 |

| | |
|------------------------------|----|
| A.5 GUI | 89 |
| A.6 Web scraping | 90 |
| A.7 Penyiapan Data | 92 |

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Makan termasuk dalam kebutuhan primer manusia yang pada umumnya dihasilkan melalui proses memasak. Proses memasak merupakan kegiatan mengolah bahan pangan menjadi makanan yang siap untuk disantap. Adanya perkembangan dalam dunia memasak menghasilkan beragam variasi menu masakan. Setiap orang memiliki selera masing-masing dalam memilih menu masakan sehingga menu kesukaan itu yang seringkali dimasak. Beragam variasi menu masakan berpotensi membuat bingung untuk memilih menu yang sesuai dengan selera dan keinginan. Selain itu, untuk memasak menu perlu memenuhi kebutuhan bahan dan bumbu baik dari yang sudah dimiliki maupun dengan membeli bahan dan bumbu yang tidak tersedia. Terkadang dengan keterbatasan manusia pemilihan menu masakan hanya terbatas pada menu yang sering dibuat dan menyebabkan pemilihan menu masakan yang repetitif. Pemilihan menu yang repetitif dapat berpotensi membuat seseorang bosan dengan menu masakan tersebut. Sehingga jangka waktu pemilihan masakan juga menjadi pertimbangan dalam menentukan pemilihan masakan. Semakin banyaknya pertimbangan tersebut maka semakin menyulitkan dalam pemilihan menu masakan. Berdasarkan masalah tersebut maka diperlukan sebuah sistem yang dapat memberikan saran atau rekomendasi menu masakan.

Informasi lengkap mengenai sebuah masakan dapat diperoleh dari resep masakan tersebut. Resep mengandung bahan, bumbu, cara pengolahan, dan informasi lain untuk menghasilkan suatu menu. Beragam bahan dan bumbu dapat membentuk menu masakan yang sangat berbeda. Oleh karena adanya beragam perpaduan bahan dan bumbu sehingga dapat membentuk resep untuk berbagai menu. Beberapa resep dilengkapi dengan informasi tambahan mengenai masakan yang membuat informasi menjadi lebih lengkap. Selain data resep dalam memilih menu masakan seseorang dapat mempertimbangkan berdasarkan kebutuhannya. Oleh karena itu seseorang dapat mengutarakan kebutuhan dan keinginan akan menu yang ingin dipilih. Pada pembangunan perangkat lunak maka pengguna dapat mengutarakan keinginan dan kebutuhannya akan menu agar diterima sistem.

Rekomendasi dapat diberikan dengan mencocokkan kebutuhan pengguna dengan informasi dari menu masakan. Dalam kasus ini dapat diimplementasikan *cosine similarity* untuk menjadi acuan skor kecocokan untuk rekomendasi. *Cosine similarity* digunakan untuk melakukan perhitungan antara dua vektor dengan mencari nilai *cosinus* dari sudut antara kedua vektor. Semakin besar skor perhitungan semakin mirip kedua vektor. *Cosine similarity* biasa digunakan untuk membandingkan dokumen[5].

Semakin berkembangnya teknologi memungkinkan sebuah perangkat lunak dapat menerima informasi langsung dari penggunanya. Hal tersebut dapat dilakukan oleh *chatbot* yang merupakan sebuah sistem yang memang didesain untuk melakukan percakapan dengan manusia. *Chatbot* didesain agar mengikuti karakteristik manusia dalam berkomunikasi sehingga dapat menerima dan memberikan *output* langsung kepada pengguna dengan bahasa seperti manusia[5]. Interaksi yang terbentuk sudah memungkinkan *input* yang bukan lagi tombol ke tombol melainkan teks berbentuk *natural language* dan suara.

Penggunaan *text-to-speech* dan *speech-to-text* sudah semakin umum diterapkan pada *gadget* untuk memudahkan manusia melakukan interaksi. Dalam hal ini dapat membantu memudahkan

interaksi untuk pengguna dalam memberikan dan menerima informasi. Google Cloud Platform menyediakan API *Text-to-Speech* yang dapat mengubah teks menjadi *natural sounding speech* atau suara dan API *Speech-to-Text* yang dapat mengubah suara menjadi teks. Dengan adanya kedua hal tersebut dapat membuat sebuah perangkat lunak menjadi interaktif dengan suara.

Pada penelitian ini, akan dibangun sebuah perangkat lunak yang mampu memberikan rekomendasi menu favorit yang sudah disesuaikan dengan keinginan pengguna. Keinginan pengguna dapat diperoleh dari *input* yang diterima perangkat lunak. Untuk mendapatkan keinginan pengguna dibangun *chatbot* yang dapat memberikan pertanyaan yang mengarahkan pengguna pada opsi-opsi parameter untuk rekomendasi menu. *Input* dan hasil rekomendasi tersebut dikeluarkan dalam bentuk suara sehingga perangkat lunak menjadi interaktif. *Chatbot* dan perangkat lunak akan dibangun menggunakan bahasa pemrograman Python. Proses konversi suara ke teks dan teks ke suara diimplementasikan dengan bantuan API Google Cloud Platform. Dibangunnya perangkat lunak tersebut diharapkan dapat membantu memberikan rekomendasi menu dan mengatasi rasa kebingungan dalam memilih menu masakan.

1.2 Rumusan Masalah

Rumusan masalah yang muncul berdasarkan deskripsi dan latar belakang yang sudah dipaparkan adalah sebagai berikut:

1. Bagaimana mengetahui kebutuhan dari perangkat lunak yang akan dibangun?
2. Bagaimana proses mengumpulkan data untuk pembentukan rekomendasi menu masakan?
3. Bagaimana mempelajari konsep penerapan *chatbot* dan *cosine similarity* untuk pembangunan perangkat lunak rekomendasi?
4. Bagaimana cara memberikan rekomendasi menu masakan kepada pengguna?
5. Bagaimana membuat proses rekomendasi mampu menerima masukan dan keluaran interaktif dengan suara?

1.3 Tujuan

Adapun tujuan penelitian dari rumusan masalah yang telah dipaparkan adalah sebagai berikut:

1. Melakukan analisis kebutuhan dengan melakukan wawancara.
2. Melakukan pengumpulan data dari *website* resep masakan yang sudah disesuaikan dengan hasil analisis kebutuhan.
3. Melakukan studi literatur mengenai *chatbot* dan *cosine similarity*.
4. Membangun perangkat lunak *chatbot* untuk memberikan rekomendasi menu masakan.
5. Mengimplementasi *text to speech* dan *speech to text* pada perangkat lunak agar perangkat lunak interaktif dengan suara.

1.4 Batasan Masalah

Batasan-batasan masalah untuk penelitian ini adalah sebagai berikut:

1. Penelitian dilakukan dengan menggunakan satu bahasa, yaitu bahasa Indonesia.
2. Menu yang direkomendasikan hanya menu makanan.
3. Perangkat lunak *chatbot* yang dikembangkan berupa perangkat lunak berbasis GUI sederhana.

1.5 Metodologi

Metodologi yang digunakan dalam penelitian ini adalah:

1. Mendefinisikan masalah yang ingin diselesaikan.
2. Melakukan studi literatur mengenai *data science*.

3. Melakukan studi literatur mengenai *chatbot*.
4. Melakukan studi literatur mengenai *natural language processing*.
5. Melakukan studi literatur mengenai *vector space model*.
6. Melakukan studi literatur mengenai *cosine similarity*.
7. Melakukan studi literatur mengenai *speech to text*.
8. Melakukan studi literatur mengenai *text to speech*.
9. Melakukan studi literatur mengenai *web scraping*.
10. Melakukan studi literatur mengenai Google Cloud Platform.
11. Melakukan analisis kebutuhan data melalui wawancara.
12. Mengumpulkan data resep dari *website* penyedia resep makanan.
13. Melakukan eksplorasi dan penyiapan data resep.
14. Melakukan analisis *chatbot*.
15. Melakukan implementasi *cosine similarity* untuk perhitungan skor rekomendasi pada *chatbot*.
16. Melakukan analisis API Google Cloud Platform untuk membuat fitur interaktif dengan suara.
17. Melakukan implementasi *speech to text* dan *text to speech* pada *chatbot*.
18. Membangun perangkat lunak.
19. Melakukan pengujian perangkat lunak.
20. Menulis dokumen skripsi.

1.6 Sistematika Pembahasan

Penelitian ini tersusun ke dalam enam bab secara sistematis sebagai berikut:

- **Bab 1: Pendahuluan**
Membahas latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, dan sistematika pembahasan.
- **Bab 2: Dasar Teori**
Membahas hasil studi literatur mengenai dasar teori yang digunakan pada penelitian ini untuk menjawab rumusan dan tujuan masalah seperti teori mengenai *chatbot*, *vector space model*, *cosine similarity*, *web scraping*, *speech to text*, *text to speech* dan Google Cloud Platform yang digunakan pada penelitian ini.
- **Bab 3: Eksplorasi Teknologi**
Membahas eksplorasi terkait teknologi yang digunakan pada penelitian ini, serta membahas penggunaan dan cara kerja teknologi yang digunakan seperti *library* yang digunakan untuk melakukan penarikan data, API Google Cloud Platform, dan *library* untuk membantu penerapan *Speech to Text* dan *Text to Speech*.
- **Bab 4: Analisis dan Rancangan**
Membahas analisis masalah, analisis kebutuhan data, ide solusi, pengumpulan dan penyiapan data, rancangan struktur data dan rancangan algoritma.
- **Bab 5: Implementasi dan Pengujian Perangkat Lunak**
Membahas implementasi perangkat lunak dan hasil pengujian perangkat lunak.
- **Bab 6: Kesimpulan dan Saran**
Membahas kesimpulan dan saran dari penelitian yang telah dilakukan.