

PENGEMBANGAN CHATBOT SEBAGAI PENGENALAN OBJEK WISATA DI SELINGKAR WILIS MENGGUNAKAN RULE-BASED

Syarifatul Ummah*¹, Dwi Rolliawati², Anang Kunaefi³, Andhy Permadi⁴

Fakultas Sains dan Teknologi Universitas Islam Negeri Sunan Ampel Surabaya

E-mail Korespondensi : syarifatum@gmail.com¹, dwi_roll@uinsa.ac.id², akunaefi@uinsa.ac.id³,
andhy@uinsa.ac.id⁴

History Artikel

Diterima : 09 Juli 2024 Disetujui : 30 September 2024 Dipublikasikan : 26 Oktober 2024

Abstract

The Selingkar Wilis area offers a variety of tourist attractions, including natural, historical, and cultural sites. However, information about these attractions is not yet fully available and easily accessible to tourists. The purpose of this study is to develop a chatbot that provides information about tourist attractions in the Selingkar Wilis area. The research object involves tourist attractions across several districts, such as Kediri, Nganjuk, Tulungagung, Trenggalek, Madiun, and Ponorogo. The research object adopts the Waterfall software development method, which includes the stages of analysis, design, implementation, and testing. Data was obtained through interviews with users and stakeholders, then designed using UML diagrams, including use case diagrams, activity diagrams, and class diagrams. The chatbot implementation using rule-based was carried out using the Django framework for the backend, MySQL as the database, and Natural Language Processing (NLP) to understand and respond to user input. The test result is shown that the chatbot was able to provide information of natural, cultural, and historical tourism. However, the chatbot still has limitations in understanding inputs unrelated to "tourism." The success rate of the tests reached 72%, with some scenarios requiring further improvement, such as handling typos and ambiguous input. Thus, this chatbot is expected to improve the user's travel experience and support tourism development in the Selingkar Wilis area. Further development is suggested, including multi-language features and the use of images as responses to increase added value in providing information.

Keywords: *Tourism, Waterfall, Unified Modeling Language (UML), Chatbot, Rule-based.*

Abstrak

Kawasan Selingkar Wilis memiliki banyak objek wisata yang menarik, termasuk wisata alam, sejarah, dan budaya. Namun, informasi mengenai wisata ini belum tersedia dengan lengkap dan mudah diakses oleh wisatawan. Tujuan penelitian ini adalah untuk mengembangkan chatbot yang dapat memberikan informasi mengenai objek wisata di sekitar Selingkar Wilis. Objek penelitian melibatkan kawasan wisata yang tersebar di beberapa kabupaten, seperti Kediri, Nganjuk, Tulungagung, Trenggalek, Madiun, dan Ponorogo. Metode penelitian yang digunakan adalah metode pengembangan perangkat lunak *Waterfall*, yang meliputi tahap analisis, desain, implementasi, dan pengujian. Data diperoleh melalui wawancara dengan pengguna dan *stakeholder*, kemudian dirancang dalam bentuk *Unified Modeling Language* (UML) diagram, termasuk use case diagram, activity diagram, dan class diagram. Implementasi chatbot menggunakan *rule-based* dilakukan dengan menggunakan *framework* Django untuk *backend* dan MySQL sebagai *database*, serta *Natural Language Processing* (NLP) untuk memahami dan merespons input pengguna. Hasil pengujian menunjukkan bahwa chatbot mampu memberikan informasi objek wisata dengan kategori wisata alam, budaya, dan sejarah. Namun, chatbot masih memiliki keterbatasan dalam memahami input yang tidak terkait dengan "wisata." Persentase keberhasilan pengujian mencapai 72%, dengan beberapa skenario yang memerlukan perbaikan lebih lanjut, seperti menangani kesalahan penulisan dan input ambigu. Dengan demikian, chatbot ini diharapkan dapat meningkatkan pengalaman wisata pengguna dan mendukung perkembangan pariwisata di kawasan Selingkar Wilis. Pengembangan lebih lanjut disarankan, termasuk fitur multi bahasa dan penggunaan gambar sebagai respon untuk meningkatkan nilai tambah dalam memberikan informasi.

Kata Kunci: Wisata, *Waterfall*, *Unified Modeling Language* (UML), Chatbot, *Rule-based*.

How to Cite: Syarifatul Ummah, dkk (2024). *Pengembangan Chatbot sebagai Pengenalan Objek Wisata di Selingkar Wilis menggunakan Rule-based*. KOMPUTEK : Jurnal Teknik Universitas Muhammadiyah Ponorogo. Vol 8 (2): Halaman 27-39

© 2024 Universitas Muhammadiyah Ponorogo. All rights reserved

ISSN 2614-0985 (Print)

ISSN 2614-0977 (Online)

PENDAHULUAN

Bandara Dhoho merupakan salah satu Proyek Strategis Nasional (PSN) yang telah beroperasi sejak awal tahun 2024. Bandara ini terletak di Kediri, Jawa Timur yang lokasinya strategis berada di lereng Gunung Wilis yang berpotensi memberikan dampak signifikan terhadap industri pariwisata di sekitar wilayah Selingkar Wilis. Dengan potensi ini, Bandara Dhoho diharapkan dapat memperkuat daya tarik pariwisata di kawasan tersebut, sekaligus menciptakan peluang baru dalam pengembangan sektor pariwisata yang tersebar di Selingkar Wilis. Kawasan Selingkar Wilis mencakup beberapa kabupaten di sekitar Bandara Dhoho, antara lain Kabupaten Kediri, Nganjuk, Tulungagung, Trenggalek, Madiun, dan Ponorogo. Kawasan ini kaya akan objek-objek wisata yang menarik, termasuk tempat wisata alam, sejarah dan budaya. Namun, informasi mengenai objek-objek wisata ini belum tersedia secara lengkap dan mudah diakses oleh wisatawan.

Seiring dengan perkembangan teknologi, terutama di bidang informasi dan komunikasi, peluang untuk meningkatkan aksesibilitas informasi wisata menjadi semakin besar. Teknologi informasi telah membuka jalan untuk memperkenalkan objek wisata secara lebih mudah dan efisien kepada publik. Namun, meskipun kemajuan teknologi telah memungkinkan penyebaran informasi secara lebih cepat dan luas, masih ada keterbatasan dalam akses informasi wisata yang memadai, dan terkadang wisatawan membutuhkan waktu yang cukup lama untuk menemukan informasi yang mereka perlukan. Hal ini menjadi perhatian penting dalam era digital, di mana kebutuhan akan informasi yang akurat, tepat, dan efisien terus meningkat, terutama di sektor pariwisata yang sedang mengalami transformasi signifikan.

Kemajuan teknologi kini telah membuat informasi menjadi lebih luas dan lebih mudah diakses di mana saja, memberikan kemudahan bagi mereka yang ingin mencari informasi. Salah satu perkembangan teknologi yang pesat saat ini adalah *Artificial Intelligence* (AI), di mana komputer dapat melakukan tugas-tugas tertentu, termasuk interaksi manusia melalui robot obrolan atau dapat disebut chatbot (Almustaqim & Toscani, 2022). Chatbot merupakan salah satu bentuk AI yang berupa program komputer yang dapat berkomunikasi dengan pengguna melalui teks (Rahayu et al., 2020). Penggunaan chatbot telah diterapkan dalam berbagai tujuan praktis, seperti menyediakan bantuan online, memberikan layanan pribadi, atau memberikan informasi yang dibutuhkan pengguna (Nugroho & Voutama, 2024).

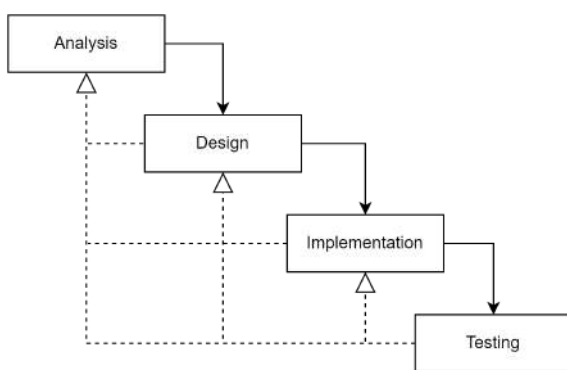
Chatbot merupakan salah satu bentuk kecerdasan buatan yang memanfaatkan teknologi *Natural Language Processing* (NLP) (Harahap & Fitria, 2020). Teknologi ini telah menjadi solusi yang populer dalam menyediakan layanan informasi interaktif. Dengan adanya chatbot, pengguna dapat mengajukan pertanyaan atau permintaan, dan mendapatkan respon yang relevan secara otomatis tanpa memerlukan intervensi dari manusia (Sindy Nova et al., 2024).

Dalam konteks pengenalan objek wisata di Selingkar Wilis, perancangan chatbot dapat membantu meningkatkan efisiensi dalam memberikan informasi kepada wisatawan. Terkadang, wisatawan sering kali merasa kebingungan atau kurang memahami informasi mengenai destinasi yang ingin mereka kunjungi, seperti lokasi, fasilitas, aktivitas, dan sebagainya. Oleh karena itu, chatbot sebagai pengenalan objek wisata di

Selingkar Wilis dapat menjadi solusi yang efektif untuk menyajikan informasi yang relevan dengan cara yang lebih mudah dipahami. Selain itu, memberikan aksesibilitas informasi yang lebih baik kepada pengguna, meningkatkan pengalaman wisata mereka, dan mendukung pertumbuhan sektor pariwisata di kawasan tersebut.

METODE PENELITIAN

Penelitian terkait perancangan chatbot sebagai pengenalan objek wisata di Selingkar Wilis ini menggunakan metode *Waterfall*, yang berfokus pada pengembangan chatbot secara bertahap dan terstruktur dalam kerangka *Software Development Life Cycle* (SDLC). SDLC berfungsi sebagai panduan yang mencakup langkah-langkah pengembangan perangkat lunak (Rabani Herdiansyah & Voutama, 2024). Adapun tahapannya, sebagaimana berikut:



Gambar 1. Tahapan Waterfall SDLC

Adapun proses dan tahapan dalam pengembangan proyek tersebut, yakni sebagai berikut:

1. *Analysis*

Analisis dilakukan untuk mengidentifikasi kebutuhan pengguna dan fitur yang diinginkan dalam chatbot yang dilakukan melalui wawancara dengan calon pengguna dan *stakeholder* (Pratama & Al Irsyadi, 2021).

Informasi yang dikumpulkan mencakup jenis pertanyaan yang sering diajukan wisatawan, serta informasi penting yang perlu disediakan, seperti lokasi, deskripsi objek wisata, dan waktu operasional.

2. *Design*

Setelah melakukan identifikasi kebutuhan, tahap berikutnya adalah merancang desain sistem, yang berupa perancangan arsitektur diagram, diagram UML dan struktur database. Pada perancangan diagram UML mencakup use case diagram, activity diagram, dan class diagram menggunakan draw.io sebagai *toolsnya*.

3. *Implementation*

Pada tahap implementasi, pengembangan kode dilakukan berdasarkan desain yang telah disusun (Huberta & Wijaya, 2023). Antarmuka dibangun dengan HTML untuk struktur dan CSS untuk tampilan. Sementara *back-end* dibangun menggunakan *Framework* Django, mengintegrasikan basis data MySQL untuk menyimpan informasi objek wisata. Fitur chatbot diimplementasikan dengan menggunakan teknik *Natural Language Processing* (NLP) untuk memahami dan merespons pertanyaan pengguna (Yuhandri et al., 2023).

4. *Testing*

Tahap pengujian dilakukan untuk memastikan bahwa sistem berfungsi dengan baik (Cahya et al., 2021). Pengujian dilakukan menggunakan berbagai skenario untuk menguji respons chatbot terhadap input pengguna, menggunakan metode *black box* untuk mengevaluasi fungsionalitas yang telah diterapkan pada chatbot (Syahrani et al., 2024).

HASIL DAN PEMBAHASAN

1. Analysis

Pada tahap awal, yakni analisis ini dilakukan identifikasi kebutuhan fungsional dan non-fungsional dari sistem yang akan dikembangkan, menyusun *flowchart* dan alur *rule-based*.

a. Hasil requirement

Hasil identifikasi kebutuhan pengguna yang telah dilakukan melalui wawancara direpresentasikan dalam tabel 1. berikut ini:

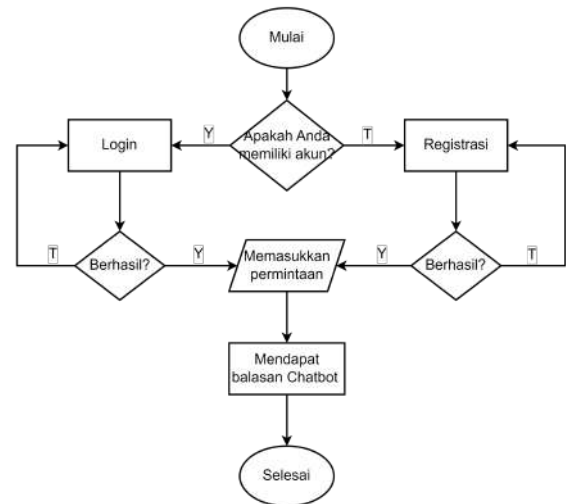
Tabel 1. Hasil *requirement*

Fitur	Deskripsi Penggunaan
Register	Pengguna baru diharuskan membuat akun dengan mengisi informasi yang diperlukan, seperti username, email, dan kata sandi. Fitur ini memungkinkan pengguna untuk mengakses layanan chatbot lebih lanjut.
Login	Pengguna yang telah memiliki akun dapat masuk ke akun mereka dengan menggunakan email dan kata sandi yang telah didaftarkan. Fitur ini memastikan bahwa hanya pengguna terautentikasi yang dapat menggunakan chatbot.
Chat	Pengguna dapat berinteraksi dengan chatbot untuk mendapatkan informasi terkait objek wisata, seperti deskripsi wisata, lokasi, dan jam operasional.
Logout	Setelah selesai menggunakan chatbot, pengguna dapat keluar dari akun mereka untuk menjaga keamanan informasi.

b. Flowchart Chatbot

Flowchart chatbot menggambarkan alur yang dilakukan sistem chatbot itu sendiri. Dalam hal ini, flowchart chatbot direpresentasikan pada gambar 2. Prosesnya dimulai dengan pengguna diarahkan untuk login. Jika belum memiliki akun, pengguna diarahkan untuk registrasi. Apabila login atau registrasi gagal, maka pengguna diminta mengisi ulang data yang diperlukan. Jika berhasil, pengguna dapat berinteraksi dengan chatbot dengan menginputkan pertanyaan atau permintaan. Setelah itu, chatbot akan

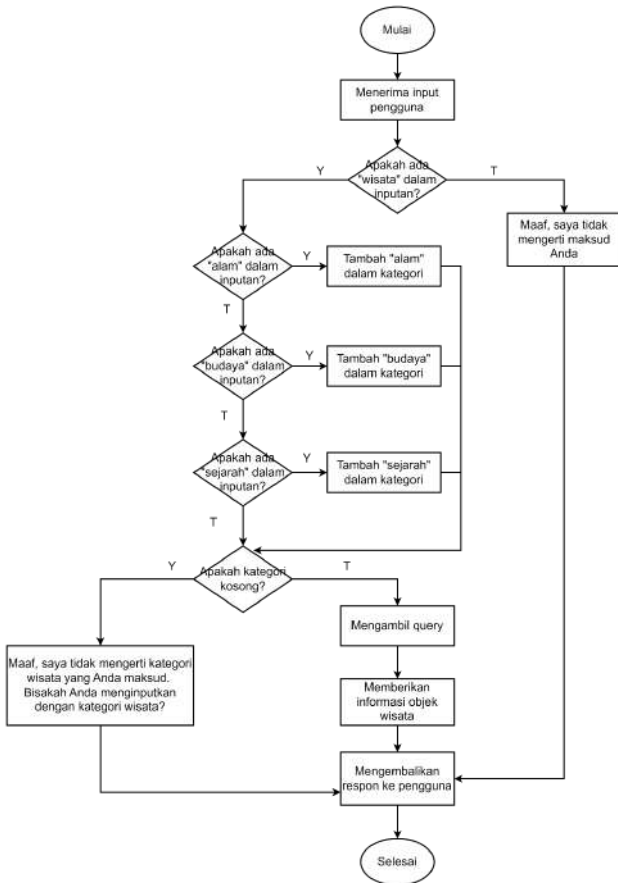
memberikan respons sesuai dengan input yang diberikan.



Gambar 2. Flowchart Chatbot

c. Alur Rule-based

Alur sistem *rule-based* pada chatbot dibuat untuk memastikan respons yang sesuai terhadap permintaan pengguna. Alur ini direpresentasikan pada gambar 3. Proses dimulai ketika chatbot menerima inputan pengguna, yang kemudian diperiksa oleh chatbot untuk mencocokkan kata kunci atau kategori yang telah ditetapkan. Kemudian, memeriksa apakah input tersebut mengandung "wisata". Jika tidak, chatbot akan memberikan respon tidak mengerti. Namun, jika Ya, chatbot akan memeriksa apakah inputannya menyebutkan kategori wisata, seperti alam, budaya, atau sejarah. Jika Ya, chatbot akan menambahkan kategori tersebut dalam kategori yang dicari. Jika Tidak, chatbot akan memeriksa apakah kategorinya kosong. Jika Tidak kosong, chatbot akan memberikan informasi objek wisata sesuai dengan kategori yang diminta. Namun, jika Ya kosong akan memberikan respon tidak menemukan informasi kategori.

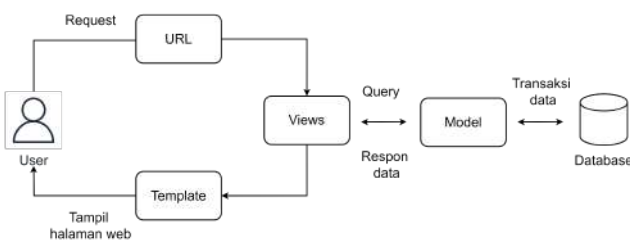


Gambar 3. Alur Rule-based Chatbot

2. Design

a. Arsitektur Diagram

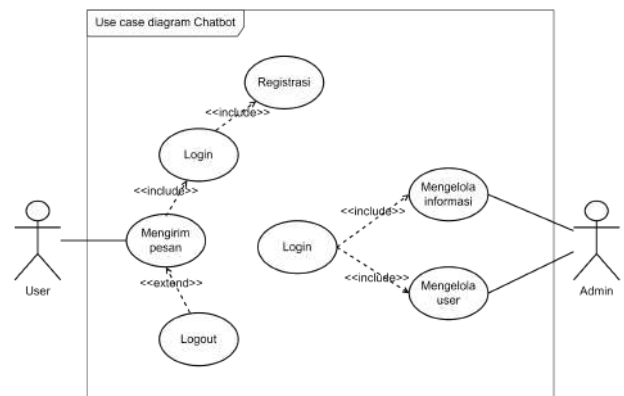
Arsitektur diagram yang digambarkan pada gambar 4. menunjukkan arsitektur sistem chatbot yang menggunakan framework Django. *User* mengirimkan permintaan ke server Django, kemudian diproses melalui *url* dan diteruskan ke *views* yang telah didefinisikan. Jika permintaan membutuhkan akses ke database, *views* akan memanggil *query* ke *models*, dan database akan mengembalikan hasilnya. Setelah proses selesai, hasilnya dipetakan ke dalam html di *template*, lalu dikirim kembali ke pengguna sebagai respon.



Gambar 4. Arsitektur Diagram

b. Use Case

Use case yang digambarkan pada gambar 5. menunjukkan peran aktor yang berinteraksi dengan chatbot. Terdapat dua aktor, yakni *user* dan *admin*. *User* adalah wisatawan, untuk berinteraksi dengan chatbot, *user* harus mendaftar terlebih dahulu, dan jika sudah memiliki akun, mereka bisa login. Sementara itu, *admin* bertugas mengelola database yang berisi informasi objek wisata dan data *user*.



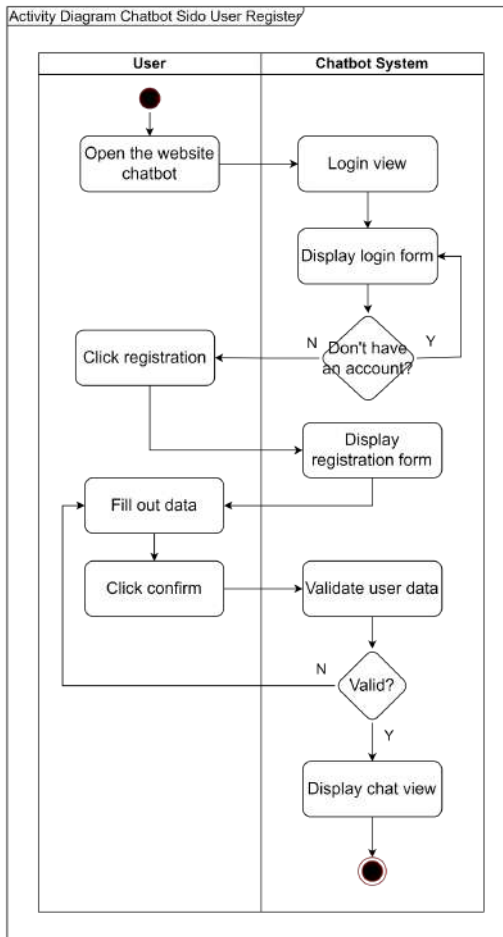
Gambar 5. Use Case

c. Activity Diagram

Activity diagram dibuat untuk menggambarkan alur aktivitas pada suatu sistem. Berikut activity diagram dari beberapa aktivitas yang telah ditetapkan:

1) Activity Diagram User Register

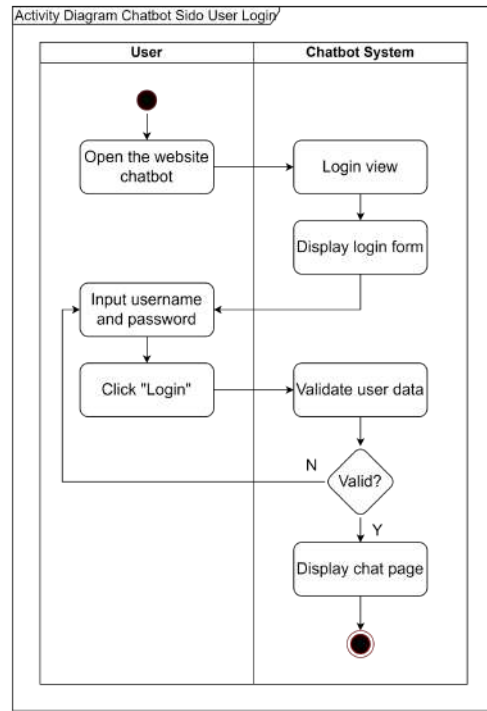
Pada aktivitas register akun, pengguna diharuskan untuk membuat akun terlebih dahulu untuk bisa berinteraksi dengan chatbot. Alur aktivitas ini direpresentasikan pada Gambar 6. Data yang harus dilengkapi pengguna adalah *username*, email, dan *password*. Setelah itu, *system* akan memvalidasi data pengguna, jika valid maka pengguna berhasil membuat akun dan dapat berinteraksi dengan chatbot. Namun, jika tidak berhasil maka pengguna tetap pada *form register* untuk mengisi data yang valid.



Gambar 6. Activity Diagram User Register

2) Activity Diagram User Login

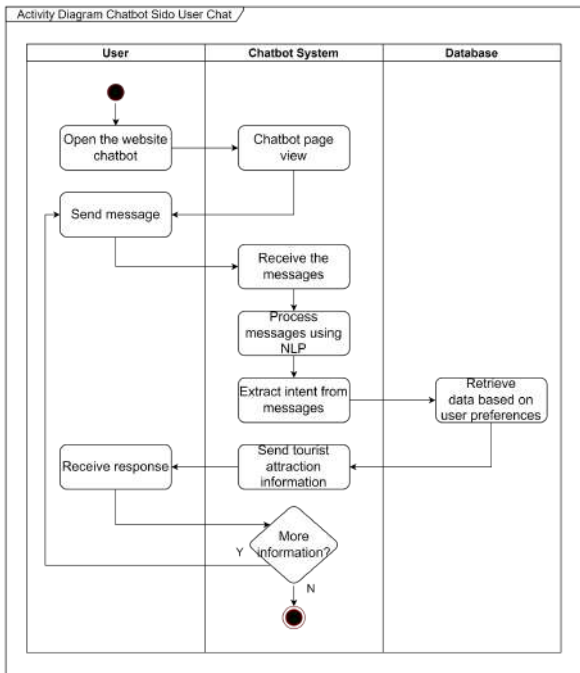
Apabila pengguna telah memiliki akun, maka pengguna hanya perlu melakukan login, yang alur aktivitasnya direpresentasikan pada gambar 7. Untuk mendapat akses masuk, pengguna diharuskan mengisi *form login* yang berisi: *username* dan *password*. Kemudian, sistem akan memvalidasi data pengguna, jika valid maka pengguna berhasil membuat akun dan dapat berinteraksi dengan chatbot. Namun, jika tidak berhasil maka pengguna tetap pada *form login* untuk mengisi data yang valid.



Gambar 7. Activity Diagram User Login

3) Activity Diagram User Chat

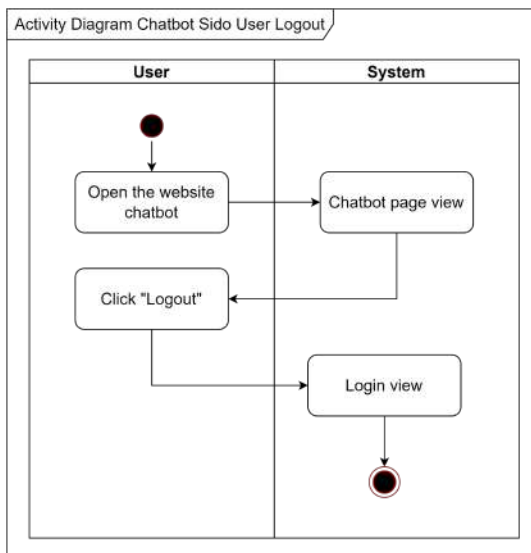
Pada aktivitas obrolan, direpresentasikan pada gambar 8. Proses dimulai saat pengguna mengirimkan pesan, kemudian chatbot memprosesnya menggunakan NLP untuk memahami maksudnya. Jika pesan pengguna relevan dengan data di database, chatbot akan mengambil dan memberikan informasi wisata yang relevan. Jika tidak, chatbot akan merespons bahwa pesan tidak dimengerti. Pengguna dapat mengirimkan pesan lagi jika ingin bertanya lebih lanjut.



Gambar 8. Activity Diagram User Chat

4) Activity Diagram User Logout

Pada gambar 9. menggambarkan aktivitas pengguna logout dari chatbot. Pada tampilan chatbot, *button logout* terletak di pojok kanan atas sehingga pengguna dapat mengklik *logout* ketika ingin keluar dari chatbot.

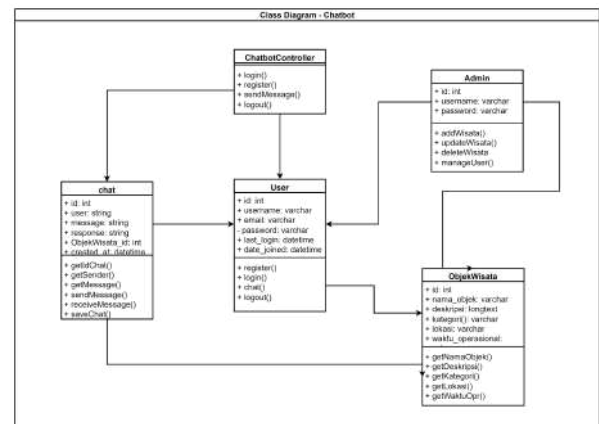


Gambar 9. Activity Diagram Logout

d. Class Diagram

Class diagram dibuat untuk menggambarkan struktur statis dari sistem

chatbot yang direpresentasikan pada gambar 10.

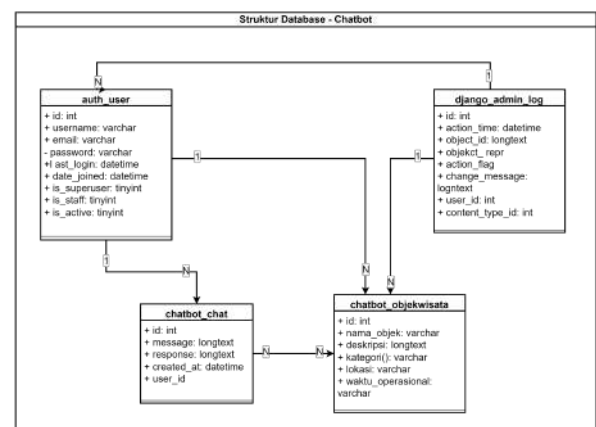


Gambar 10. Class Diagram

Pada kelas “ChatbotController” merupakan komponen utama yang mengelola interaksi pengguna dan pesan obrolan. Pengguna dapat berinteraksi dengan chatbot untuk mendapatkan informasi objek wisata, sementara admin dapat mengelola system melalui class admin. Kelas “chat” menangani pengiriman pesan antara pengguna dan chatbot.

e. Entity Relationship Diagram (ERD)

ERD dibuat untuk menggambarkan struktur database, yang direpresentasikan pada gambar 11.



Gambar 11. Entity Relationship Diagram

Terdapat empat tabel utama: *auth_user*, *chatbot_chat*, *chatbot_objekwisata*, dan *django_admin_log*. Tabel *auth_user* menyimpan informasi pengguna. Tabel *chatbot_chat* mencatat percakapan antara

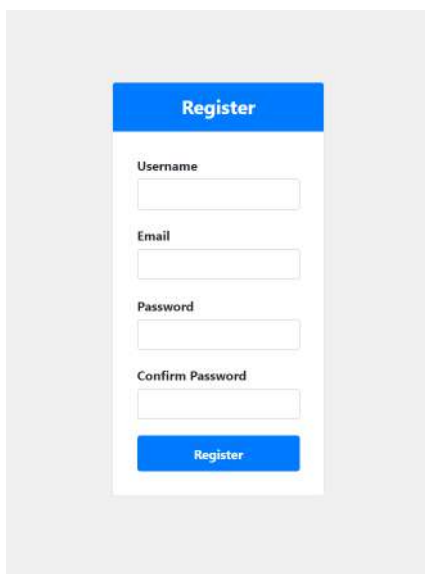
pengguna dan chatbot, termasuk pesan dan respons. Tabel chatbot_objekwisata menyimpan informasi tentang objek wisata. Tabel django_admin_log mencatat aktivitas administrasi, termasuk perubahan data dan tindakan lainnya.

3. Implementation

a. Tampilan Chatbot

1) Halaman Register User

Pada halaman register pengguna, merupakan awal bagi pengguna yang belum memiliki akun pada chatbot sehingga mengharuskan pengguna untuk membuat akun terlebih dahulu, sebagaimana ditampilkan pada gambar 12. Adapun data yang harus dilengkapi, yakni *username*, email dan *password*. System akan memverifikasi data registrasi dan jika datanya valid, akunnya akan berhasil terdaftar.

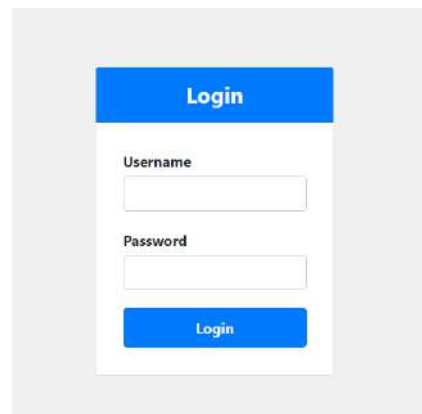


Gambar 12. Tampilan Register User

2) Halaman Login User

Halaman login merupakan pintu masuk utama bagi pengguna. Apabila pengguna sudah memiliki akun, dapat melakukan login, namun jika belum, pengguna dapat klik Register agar diarahkan untuk membuat akun terlebih dahulu. Sebagaimana ditampilkan

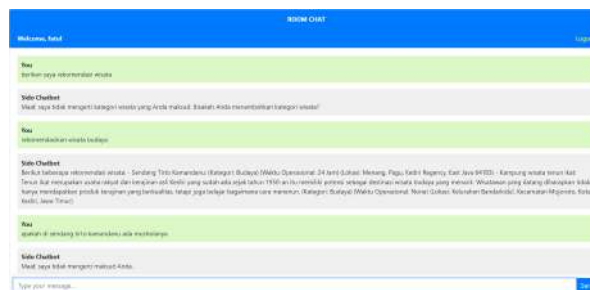
pada gambar 13., pengguna diminta untuk memasukkan *username* dan *password* yang telah mereka daftarkan sebelumnya.



Gambar 13. Tampilan Login User

3) Halaman Room Chat User

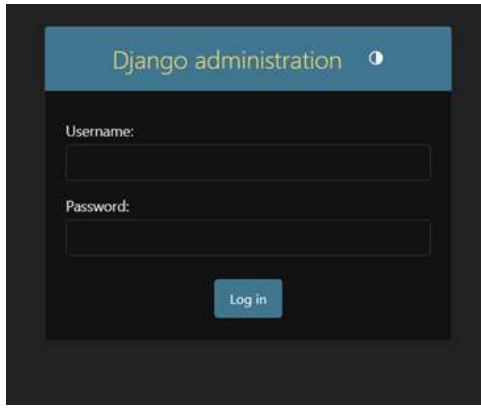
Room chat adalah tempat di mana pengguna dapat berinteraksi langsung dengan Chatbot. Sebagaimana ditampilkan pada gambar 14. Pada halaman ini, pengguna dapat mengajukan pertanyaan dan menerima informasi mengenai objek wisata.



Gambar 14. Tampilan Room Chat

4) Halaman Login Admin

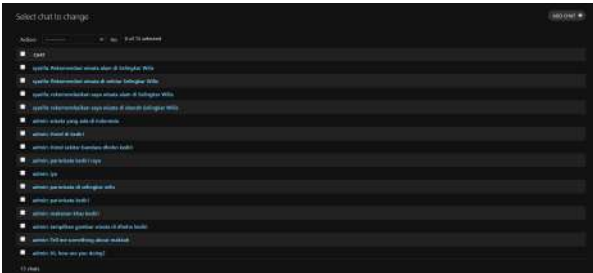
Halaman login pada admin yang ditampilkan pada gambar 15., dirancang khusus oleh framework Django. Jadi, tidak perlu merancang halaman untuk admin karena Django telah menyediakan antarmukanya.



Gambar 15. Tampilan Login Admin

5) Halaman Database Admin

Pada halaman database admin yang ditampilkan pada gambar 16. dapat mengelola data pengguna dan data objek wisata. Halaman ini memberikan akses penuh kepada admin untuk menambah, mengubah, atau menghapus informasi yang disimpan dalam database sistem.



Gambar 16. Tampilan Database Admin

4. Testing

a. Pengujian Register

Tabel 2. Pengujian Register

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Input data yang benar	Registrasi berhasil, menampilkan halaman chat	Registrasi berhasil, menampilkan halaman chat	Berhasil
Input password yang berbeda pada field confirm password	Menampilkan pesan "Password don't match"	Menampilkan pesan "Password don't match"	Berhasil

Tidak mengisi data dengan lengkap	Menampilkan pesan "Please fill out this field"	Menampilkan pesan "Please fill out this field"	Berhasil
Mengisi field username dengan karakter special (! @ # \$ % ^ & *)	Menolak registrasi dan menampilkan pesan "Invalid Username"	Registrasi berhasil dan menampilkan halaman chat	Tidak berhasil
Mengisi field email dengan format yang tidak valid, misalnya "username @"	Menolak registrasi dan menampilkan pesan "Invalid Email"	Registrasi berhasil dan menampilkan halaman chat	Tidak berhasil

Berdasarkan tabel 2. pengujian fitur registrasi menunjukkan bahwa 3 dari 5 skenario berhasil, sisanya tidak. Validasi pada form registrasi sebagian besar telah berjalan dengan baik, tetapi tidak berhasil menangani karakter khusus di field username, yang seharusnya ditolak. Selain itu, validasi email dengan format yang tidak valid juga gagal, karena sistem tidak menolak pendaftaran dan langsung menampilkan halaman chat.

b. Pengujian Login

Tabel 3. Pengujian Login

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menginputkan data yang benar pada form login	Registrasi berhasil, menampilkan halaman chat	Registrasi berhasil, menampilkan halaman chat	Berhasil
Menginputkan password yang salah	Menampilkan pesan "Invalid username or password"	Menampilkan pesan "Invalid username or password"	Berhasil

Tidak mengisi data dengan lengkap	Menampilkan pesan "Please fill out this field"	Menampilkan pesan "Please fill out this field"	Berhasil
Menginputkan username yang belum terdaftar	Menampilkan pesan "Invalid username or password"	Menampilkan pesan "Invalid username or password"	Berhasil

Berdasarkan tabel 3. pengujian fitur login menunjukkan bahwa seluruh skenario validasi berjalan dengan baik dan sesuai harapan. Sistem berhasil merespons dengan tepat terhadap data yang benar, password yang salah, kelengkapan data, dan username yang belum terdaftar.

c. Pengujian Obrolan

Tabel 4. Pengujian Obrolan

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Kirim pesan dengan pertanyaan/pemintaan seputar wisata, misalnya "beri saya rekomendasi wisata"	Maaf, saya tidak mengerti kategori wisata yang Anda maksud. Bisakah Anda menginputkan dengan kategori wisata?	Maaf, saya tidak mengerti kategori wisata yang Anda maksud. Bisakah Anda menginputkan dengan kategori wisata?	Berhasil
Kirim pesan tanpa "wisata", misalnya "apakah di Kebun Rambutan ada musholanya"	Maaf, saya tidak mengerti maksud Anda.	Maaf, saya tidak mengerti maksud Anda.	Berhasil
Kirim pesan dengan pertanyaan/pemintaan	Memberikan informasi wisata alam	Memberikan informasi wisata alam	Berhasil

seputar wisata berdasarkan kategori wisata alam			
Kirim pesan dengan pertanyaan/pemintaan seputar wisata berdasarkan kategori wisata budaya	Memberikan informasi wisata budaya	Memberikan informasi wisata budaya	Berhasil
Kirim pesan dengan pertanyaan/pemintaan seputar wisata berdasarkan kategori wisata sejarah	Memberikan informasi wisata sejarah	Memberikan informasi wisata sejarah	Berhasil
Kirim pesan "rekomendasi wisata alam dan budaya"	Memberikan informasi wisata alam dan budaya	Memberikan informasi wisata alam dan budaya	Berhasil
Kirim pesan "rekomendasi wisata dan budaya"	Memberikan informasi wisata sendiri, budaya sendiri	Memberikan informasi wisata alam dan budaya	Tidak berhasil
Kirim pesan dengan penulisan yang tidak benar, misalnya "rekomendasi wisata alam"	Memberikan informasi wisata alam	Maaf, saya tidak mengerti maksud Anda.	Tidak berhasil
Kirim pesan "beri saya 3 wisata alam"	Memberikan informasi 3 wisata alam	Memberikan semua daftar informasi wisata alam	Tidak berhasil

Berdasarkan tabel 4. pengujian pada fitur chat menunjukkan bahwa 6 dari 9 skenario

skenario berhasil, dengan sistem mampu merespons pertanyaan tentang wisata dengan tepat, termasuk kategori alam, budaya, dan sejarah. Sistem juga dapat menampilkan pesan yang sesuai untuk input yang tidak lengkap atau tidak relevan. Namun, 3 dari 9 skenario tidak berhasil, seperti permintaan ambigu, misalnya "rekomendasikan wisata dan budaya" tidak memberikan informasi yang terpisah, tetapi memberikan informasi kombinasi. Selain itu, ketika mengirimkan pesan dengan penulisan yang salah atau *typo* seperti "rekomendasikan wisata alam", sistem tidak dapat memahami maksudnya. Skenario permintaan untuk tiga rekomendasi wisata alam menghasilkan semua informasi wisata alam, bukan tiga pilihan. Hasil pengujian ini menunjukkan bahwa perbaikan diperlukan untuk menangani permintaan ambigu, kesalahan ketik, dan permintaan spesifik.

Secara keseluruhan, skenario pengujian yang dilakukan sebanyak 18 dan terdapat 5 pengujian yang gagal. Setelah melalui tahap pengujian, didapatkan nilai persentase keberhasilan sebesar 72%. Evaluasi ketepatan dapat dijelaskan oleh rumus berikut ini

$$\begin{aligned} \text{Persentase Berhasil} &= \frac{\text{Total respon tepat}}{\text{Total skenario}} \times 100\% \\ &= \frac{13}{18} \times 100\% \\ &= 72\% \end{aligned}$$

KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan, dapat disimpulkan bahwa dari total 18 skenario yang diuji, 13 skenario berhasil dijalankan dengan baik, dengan persentase keberhasilan chatbot sebesar 72% yang artinya sebagian fungsionalitas aplikasi telah berjalan dengan baik. Namun, masih terdapat beberapa fungsi yang memerlukan perbaikan lebih lanjut. Dalam hal ini, chatbot dapat memberikan informasi objek wisata berdasarkan kategori wisata yang diinginkan

pengguna, seperti wisata alam, sejarah, atau budaya. Namun, chatbot hanya mampu memberikan respon ketika inputan pengguna mengandung "wisata" dan kategori wisata. Oleh karena itu, kemampuannya terbatas dalam menangani pertanyaan atau permintaan pengguna jika tidak relevan.

Adapun saran yang ditujukan untuk meningkatkan pengembangan chatbot, yakni sebagai berikut:

1. Membuat Chatbot dengan multibahasa sehingga dapat digunakan juga oleh wisatawan internasional.
2. Selain itu, mengembangkan dengan memberikan respon selain teks, yakni berupa gambar guna meningkatkan nilai tambah sehingga pengguna akan lebih mudah mengenali objek wisata yang telah direkomendasikan
3. Mengembangkan fungsional chatbot dengan kemampuan menerima dan merespons pertanyaan atau permintaan pengguna terkait informasi selain kategori wisata, seperti waktu operasional, lokasi, atau harga tiket suatu wisata.

DAFTAR PUSTAKA

- Almustaqim, A., & Toscany, A. N. (2022). Perancangan Sistem Chatbot Sebagai Virtual Assistant Pada Pt. Everbright Jambi. *SKANIKA: Sistem Komputer dan Teknik Informatika*, 5(2), 228–239. <https://doi.org/10.36080/skanika.v5i2.2953>
- Cahaya, N., Triayudi, A., & Benrahman, B. (2021). Implementasi Framework Codeigniter Pada Perancangan Chatbot Interaktif Menerapkan Metode Waterfall. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 5(1), 273. <https://doi.org/10.30865/mib.v5i1.2623>
- Harahap, D. W., & Fitria, L. (2020). Aplikasi Chatbot Berbasis Web Menggunakan Metode Dialogflow. *J-ICOM - Jurnal Informatika dan Teknologi Komputer*,

- 1(1), 6–13. <https://doi.org/10.33059/j-icom.v1i1.2796>
- Huberta, B., & Wijaya, A. B. (2023). Perancangan Chatbot Website Program Studi Informatika Menggunakan Framework Codeigniter. *Jurnal Informatika dan Teknik Elektro Terapan*, 11(3). <https://doi.org/10.23960/jitet.v11i3.3225>
- Nugroho, I. S., & Voutama, A. (2024). Implementasi Chat Bot Untuk Pelayanan Pelanggan Yang Terintegrasi Web Toko Komputer. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(3), 3132–3136. <https://doi.org/10.36040/jati.v8i3.9630>
- Pratama, N. Y. N., & Al Irsyadi, F. Y. (2021). Perancangan Chatbot Islami untuk Aplikasi ChatAja. *Emitor: Jurnal Teknik Elektro*, 21(1), 64–71. <https://doi.org/10.23917/emitor.v21i1.12123>
- Rabani Herdiansyah, H., & Voutama, A. (2024). Implementasi Chatbot Pada Aplikasi Sewa Kamera Berbasis Website Menggunakan Metode Waterfall. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(4), 4473–4477. <https://doi.org/10.36040/jati.v8i4.9910>
- Rahayu, D., Mukrodin, M., & Hariyono, R. (2020). Penerapan Artificial Intelligence Dalam Aplikasi Chatbot Sebagai Helpdesk Objek Wisata Dengan Permodelan Simple Reflex-Agent (Studi Kasus: Desa Karangbenda). *Smart Comp :Jurnalnya Orang Pintar Komputer*, 9(1), 7–21. <https://doi.org/10.30591/smartcomp.v9i1.1813>
- Sindy Nova, Nurul Khotimah, & Maria Y Aryati Wahyuningrum. (2024). Pemanfaatan Chatbot Menggunakan Natural Language Processing Untuk Pembelajaran Dasar-Dasar Gui Tkinter Pada Bahasa Pemrograman Python. *Jurnal Ilmiah Teknik*, 3(1), 58–65. <https://doi.org/10.56127/juit.v3i1.1162>
- Syahrani, G., Sevira, S., & Yunizar Pratama Yusuf, A. (2024). Rancangan Chatbot Rekomendasi Coffee Shop Jabodetabek dengan Menggunakan Dialogflow Natural Language Processing. *SKANIKA: Sistem Komputer dan Teknik Informatika*, 7(1), 74–84. <https://doi.org/10.36080/skanika.v7i1.3139>
- Yuhandri, Y., Sovia, R., Syaifullah, A., Yenila, F., & Permana, R. (2023). Penerapan Natural Language Processing Pada Sistem Chatbot Sebagai Helpdesk Obyek Wisata Menggunakan Metode Naïve Bayes. *Jurnal Infortech*, 5(2), 210–218. <https://doi.org/10.31294/infortech.v5i2.20911>