



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

JUDUL

**PENGEMBANGAN *CHATBOT* PADA INFORMASI
LAYANAN BERITA BERBASIS WEB MENGGUNAKAN
*FRAMEWORK REACT***

TUGAS AKHIR

RONI PRAWIJAYA

0110220198

PROGRAM STUDI TEKNIK INFORMATIKA

DEPOK

AGUSTUS 2024



**STT TERPADU
NURUL FIKRI**

SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

JUDUL

**PENGEMBANGAN *CHATBOT* PADA INFORMASI
LAYANAN BERITA BERBASIS WEB MENGGUNAKAN
*FRAMEWORK REACT***

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana
Komputer**

RONI PRAWIJAYA

0110220198

STT - NF

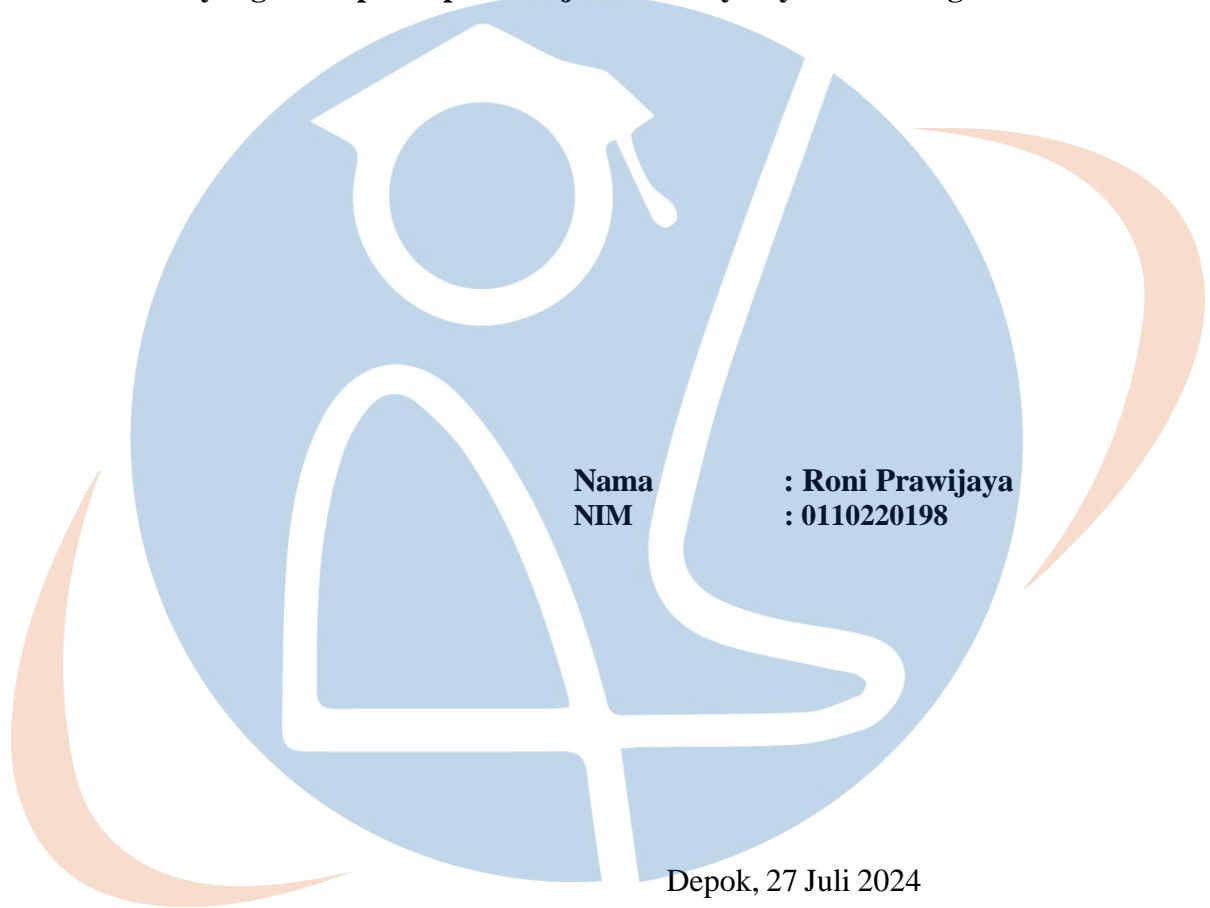
PROGRAM STUDI TEKNIK INFORMATIKA

DEPOK

AGUSTUS 2024

HALAMAN PERNYATAAN ORISINALITAS

Skripsi/Tugas Akhir ini adalah hasil karya penulis, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.



Nama : Roni Prawijaya
NIM : 0110220198

Depok, 27 Juli 2024

STT - NE
Tanda Tangan
Roni Prawijaya

Roni Prawijaya

HALAMAN PENGESAHAN

Skripsi/Tugas Akhir ini diajukan oleh :

Nama : Roni Prawijaya

NIM : 0110220198

Program Studi : Teknik Informatika

Judul Skripsi : Pengembangan *Chatbot* Pada Informasi Layanan Berita Berbasis Web Menggunakan *Framework React*.

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

DEWAN PENGUJI

Pembimbing



Henry Saptono, S.Si., M.Kom.

Penguji



Salman El Farisi S.Kom., M.Kom.

STT - NF

Ditetapkan di : Depok

Tanggal : 27 Juli 2024

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, karena atas berkat dan rahmatNya, penulis dapat menyelesaikan skripsi/Tugas Akhir ini. Penulisan skripsi/Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana komputer Program Studi Teknik Informatika pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi/tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Allah SWT.
2. Orang tua dan saudara kandung saya Rifa'i Azar, Lami, Abdul Hadi, Khobari, Sutiah, Barrurrohim, Rizki Adi Pradana, Nurul Jannah dan khoirotn Nisa anggota yang telah memberikan dorongan baik secara moril maupun materil dalam penyelesaian tugas ini.
3. Bapak Dr. Lukman Rosyidi selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Ibu Tifani Nabarian, S.Kom, M.T.I selaku Ketua Program Studi Teknik Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
5. Ibu Nurul Jannah, S.IIP., M.Hum. selaku Dosen Pembimbing Akademik yang telah membimbing penulis selama berkuliah di Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
6. Bapak Henry Saptono, S.Si, M.Kom selaku Dosen Pembimbing Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
7. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing penulis dalam menuntut ilmu yang telah diberikan.
8. Teman-teman dan wanita ciseeng yang telah memberikan semangat serta dukungan untuk bisa menyelesaikan tugas akhir ini.

Dalam penulisan ilmiah ini tentu saja masih banyak terdapat kekurangan-kekurangan yang mungkin disebabkan oleh keterbatasan kemampuan dan pengetahuan yang penulis miliki. Walaupun demikian, penulis telah berusaha menyelesaikan penulisan ilmiah ini sebaik mungkin. Oleh karena itu apabila terdapat kekurangan di dalam penulisan ilmiah ini, dengan rendah hati penulis menerima kritik dan saran dari pembaca.

Akhir kata, penulis berharap Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 27 Juli 2024



Roni Prawijaya



STT - NF

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri,
saya yang bertanda tangan di bawah ini:

Nama : Roni Prawijaya

NIM 0110220198

Program Studi : Teknik Informatika

Jenis karya : Tugas Akhir

demikian demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT NF **Hak Bebas Royalti Non Eksklusif (*Non-exclusive Royalty - Free Right*)** atas karya ilmiah saya yang berjudul :
PENGEMBANGAN *CHATBOT* INFORMASI LAYANAN BERITA BERBASIS WEB MENGGUNAKAN *FRAMEWORK REACT*

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non Eksklusif ini STT-NF berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 27 Juli 2024

Yang Menyatakan



Roni Prawijaya

ABSTRAK

Nama : Roni Prawijaya
NIM : 0110220198
Program Studi : Teknik Informatika
Judul : Pengembangan chatbot pada informasi layanan berita berbasis web menggunakan *framework react*

Pemanfaatan framework React dalam pengembangan chatbot penyedia informasi berita berbasis web membuka peluang besar untuk meningkatkan aksesibilitas masyarakat terhadap informasi berita. Penelitian ini bertujuan mengembangkan chatbot penyedia informasi berita berbasis web menggunakan framework React. Metodologi penelitian mengadopsi pendekatan R&D (Research and Development), meliputi tahapan kajian literatur, analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan evaluasi. Sistem chatbot yang dirancang mengintegrasikan React.js sebagai kerangka kerja frontend dan NewsAPI sebagai penyedia konten. Fitur utama mencakup autentikasi pengguna, antarmuka chat yang interaktif, kategori pesan, pencarian dengan kata kunci. Pengujian black box menunjukkan tingkat keberhasilan 100% untuk seluruh fitur utama, mengindikasikan reliabilitas sistem. Namun, ditemukan beberapa keterbatasan seperti ketergantungan pada konektivitas dan API pihak ketiga.

Penerapan React memungkinkan pengembangan antarmuka yang responsif dan efisien, meningkatkan performa dan pengalaman pengguna. Chatbot yang dikembangkan menggambarkan potensi untuk meningkatkan aksesibilitas dan personalisasi layanan berita, dengan menawarkan alternatif yang interaktif terhadap metode distribusi berita. Penelitian ini berhasil mencapai tujuannya dalam merancang dan mengimplementasikan chatbot layanan berita yang efektif, sambil mengidentifikasi peluang untuk pengembangan lebih lanjut. Kontribusi penelitian ini signifikan dalam konteks perkembangan teknologi informasi dan transformasi industri berita digital. Sampai pada tahap penyelesaian peneliti memberikan kesimpulan, pada penelitian ini berhasil mencapai tujuannya dalam merancang dan mengimplementasikan chatbot layanan berita berbasis web, dengan harapan adanya ruang untuk penyempurnaan dan pengembangan lebih lanjut di masa depan.

Kata kunci : Chatbot, Web, Berita, React, Teknologi.

ABSTRACT

Name : Roni Prawijaya

NIM 0110220198

Study Program : Informatics Engineering

Title : Chatbot development on web-based news service information using the React framework

The utilization of the React framework in the design of a web-based news information provider chatbot opens up great opportunities to increase public accessibility to news information. This study aims to develop a web-based news information provider chatbot using the React framework. The research methodology adopts the R&D (Research and Development) approach, including the stages of literature review, needs analysis, system design, implementation, testing, and evaluation. The designed chatbot system integrates React.js as the frontend framework and NewsAPI as the content provider. Key features include user authentication, interactive chat interface, message categorization, keyword search. Black box testing showed a 100% success rate for all major features, indicating the reliability of the system. However, there were some limitations such as dependency on third-party connectivity and API.

The implementation of React enabled the development of a responsive and efficient interface, improving performance and user experience. The developed chatbot illustrates the potential to improve the accessibility and personalization of news services, by offering an interactive alternative to news distribution methods. This research successfully achieved its goal of designing and implementing an effective news service chatbot, while identifying opportunities for further development. The contributions of this research are significant.

Key words: Chatbot, Web, News, React, Technology.

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	2
HALAMAN PENGESAHAN	3
KATA PENGANTAR	4
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS	6
ABSTRAK	7
ABSTRACT	8
DAFTAR ISI	9
DAFTAR GAMBAR	11
DAFTAR TABEL	12
BAB I	13
PENDAHULUAN	13
1.1 <i>Latar belakang</i>	13
1.2 <i>Rumusan Masalah</i>	16
1.3 <i>Tujuan Penelitian</i>	16
1.4 <i>Manfaat Penelitian</i>	16
1.5 <i>Batasan Masalah</i>	17
1.6 <i>Sistematika Penulisan</i>	17
BAB II	18
KAJIAN LITERATUR	18
2.1 <i>Website</i>	18
2.2 <i>Chatbot</i>	18
2.3 <i>Framework React</i>	19
2.4 <i>API</i>	20
2.5 <i>NPM</i>	20
2.6 <i>BlackBox</i>	21
2.7 <i>Penelitian Terkait</i>	22
BAB III	24
METODOLOGI PENELITIAN	24
3.1 <i>Tahapan Penelitian</i>	24
3.1.1 <i>Studi Penelitian</i>	25
3.1.2 <i>Analisis Kebutuhan</i>	25
3.1.3 <i>Perancangan Sistem</i>	25
3.1.4 <i>Implementasi</i>	25
3.1.5 <i>Pengujian</i>	25
3.1.6 <i>Evaluasi Hasil</i>	25
3.2 <i>Rancangan Penelitian</i>	26

3.2.1 Jenis Penelitian.....	26
3.2.2 Metode Analisis.....	26
3.2.3 Metode Pengumpulan Data	26
3.2.4 Waktu Penelitian	26
3.2.5 Lingkungan Pengembangan.....	26
3.2.6 Metode Pengujian	27
BAB IV	29
IMPLEMENTASI DAN EVALUASI	29
<i>4.1 Analisis dan Perancangan.....</i>	<i>29</i>
4.1.1 Analisis Sistem.....	29
4.1.2 Kebutuhan Sistem	30
4.1.3 Perancangan sistem	32
4.1.4 Arsitektur Sistem.....	32
4.1.5 Use Case Diagram.....	33
4.1.6 Activity Diagram.....	35
4.1.7 Perancangan UI.....	36
4.1.9 Perancangan Pengujian.....	37
<i>4.2 Implementasi Sistem.....</i>	<i>39</i>
<i>4.3 Implementasi Antarmuka</i>	<i>51</i>
4.3.1 Halaman Login.....	51
4.3.2 Halaman chat login tamu / guest.....	52
4.3.3 User memilih kategori yang disediakan oleh bot via tombol CTA (Call to Action).....	52
4.3.4 Tampilan ketika user memilih kategori custom dari tombol "Kategori Lainnya"	53
4.3.5 Tampilan jika berita tidak sesuai/tidak ada.....	53
4.3.6 Tampilan ketika keyword/prompt yang diketik user tidak sesuai	54
4.3.7 Tampilan Hasil Pencarian Berita	55
4.3.8 Tampilan Logout.....	55
<i>4.4 Evaluasi dan Pengujian.....</i>	<i>56</i>
4.4.1 Evaluasi Sistem	56
4.4.2 Pengujian Sistem.....	57
BAB V.....	58
KESIMPULAN DAN SARAN	58
<i>5.1 Kesimpulan.....</i>	<i>58</i>
<i>5.2 Saran</i>	<i>59</i>
DAFTAR REFERENSI.....	60

DAFTAR GAMBAR

Gambar 3. 1 Tahapan Penelitian.....	25
Gambar 4. 1 Perancangan Sistem	32
Gambar 4. 2 Arsitektur Sistem	33
Gambar 4. 3 Use Case Diagram.....	34
Gambar 4. 4 Activity Diagram	35
Gambar 4. 5 Wireframe Login.....	36
Gambar 4. 6 Wireframe Chat.....	37
Gambar 4. 7 Perancangan Pengujian	38
Gambar 4. 8 Implementasi Sistem.....	39
Gambar 4. 9 Instalasi Node.js	40
Gambar 4. 10 Installer Node.js	40
Gambar 4. 11 Tampilan Memilih Install	41
Gambar 4. 12 Instalasi Node.js Selesai.....	41
Gambar 4. 13 Instalasi React JS+Vite	42
Gambar 4. 14 Instalasi Vite Selesai	42
Gambar 4. 15 Jalankan npm run dev	43
Gambar 4. 16 Instalasi React.js + Vite Berhasil	43
Gambar 4. 17 Pembuatan Key API.....	44
Gambar 4. 18 Pembuatan Key API.....	44
Gambar 4. 19 Pembuatan Key API.....	45
Gambar 4. 20 Persiapan File Config API	45
Gambar 4. 21 Persiapan Halaman Login	47
Gambar 4. 22 Persiapan Halaman Dashboard Chat.....	48
Gambar 4. 23 Halaman Login.....	51
Gambar 4. 24 Halaman Chat Login Tamu.....	52
Gambar 4. 25 Halaman User Memilih Kategori	52
Gambar 4. 26 Halaman User Memilih Kategori Custom	53
Gambar 4. 27 Halaman Tampilan Jika Berita tidak Sesuai/Tidak Ada	53
Gambar 4. 28 Halaman Tampilan Ketika Keyword Tidak Sesuai	54
Gambar 4. 29 Halaman Tampilan Hasil Pencarian Berita.....	55
Gambar 4. 30 Halaman Tampilan Logout	55

STT - NF

DAFTAR TABEL

Tabel 2.1 Penelitian Terkait.....	22
Tabel 3.1 Waktu Penelitian.....	26
Tabel 3.2 Spesifikasi Perangkat Keras	27
Tabel 3.3 Template Tabel Pengujian	28
Tabel 4.1 Spesifikasi Perangkat Keras.....	31
Tabel 4.2 Script File API.js	46
Tabel 4.3 File Login.jsx	47
Tabel 4.4 Fungsi pengecekan sesi.....	48
Tabel 4.5 Fungsi pengecekan sesi.....	48
Tabel 4.6 Fungsi Current Step	49
Tabel 4.7 Step Filtering	49
Tabel 4.8 Fungsi	50
Tabel 4.9 Black Box Testing	57

STT - NF

BAB I

PENDAHULUAN

Bab akan menjelaskan terkait topik yang akan dibahas dalam penelitian ini mulai dari latar belakang, rumusan masalah, tujuan dan manfaat dari penelitian, serta batasan masalah apa saja yang akan dipaparkan pada penelitian ini.

1.1 Latar belakang

Kemajuan teknologi yang luar biasa di masa sekarang membawa perubahan besar dalam kehidupan manusia. Di era Revolusi Industri 4.0 ini, internet menjadi platform utama untuk komunikasi dan pertukaran informasi. Salah satu teknologi kunci yang akan mendorong implementasi industri 4.0 adalah kecerdasan buatan atau *Artificial intelligence (AI)* [1]. *Artificial intelligence (AI)* telah merevolusi cara kita beraktivitas sehari-hari, dimana kita menggunakan aplikasi dan perangkat canggih yang disebut *intelligent agents* yaitu sebuah sistem agen cerdas yang dirancang untuk dapat melakukan berbagai fungsi sesuai dengan kebutuhan. *Chatbot* adalah program kecerdasan buatan dan model Interaksi Manusia komputer atau *human computer interaction (HCI)* [2].

Pada tahun 1950, Alan Turing merumuskan *Turing Test*, sebuah metode inovatif berbasis percakapan untuk mengevaluasi kemampuan mesin dalam meniru manusia. *Turing Test* berfokus pada kemampuan program komputer dalam meniru percakapan manusia untuk menentukan kecerdasannya. Deskripsi Turing tentang perilaku mesin cerdas menjadi landasan konsep *chatbot* yang dikenal saat ini [3]. Teknologi *chatbot* merupakan program aplikasi AI yang dirancang untuk berinteraksi langsung dengan manusia melalui teks, suara, atau kombinasi keduanya. *Chatbot* memanfaatkan *Natural Language Processing (NLP)* dan analisis sentimen untuk berkomunikasi dalam bahasa alami, dan memungkinkan pengguna berkomunikasi dengan komputer menggunakan bahasa sehari-hari, layaknya bercakap-cakap dengan manusia [2].

Ada tiga model klasifikasi yang umum digunakan dalam pengembangan *chatbot*, yaitu *rule-based model*, *retrieval based model*, and *generative model*.

Model-model ini menghasilkan *respons* yang sesuai berdasarkan metode pemrosesan input dan pembangkitan *respons* yang diterapkan. Dalam penelitian ini, model yang akan digunakan adalah model berbasis pengambilan (*Retrieval-based model*). Model ini menawarkan fleksibilitas yang lebih tinggi karena memanfaatkan API untuk menanyakan dan menganalisis sumber daya yang tersedia. *Chatbot* berbasis *retrieval based* model mengambil beberapa kandidat *respons* dari indeks dan menerapkan pendekatan pencocokan untuk memilih *respons* yang paling informatif dan bermanfaat bagi pengguna [4].

Penerapan *chatbot* di bidang industri dan bisnis telah menunjukkan kemajuan yang signifikan, di mana *chatbot* berperan aktif dalam memberikan layanan *customer service* dan berinteraksi langsung dengan konsumen. *Chatbot* diprogram untuk melayani percakapan dengan pengguna secara responsif, memungkinkan mereka mendapatkan balasan pesan dengan cepat dan efisien [5]. Pemanfaatan kecerdasan buatan (AI) telah merevolusi dunia bisnis, menghadirkan solusi efektif untuk meningkatkan efisiensi dan profitabilitas. *Chatbot*, sebagai solusi inovatif, mampu menghemat biaya layanan dan menangani banyak pelanggan secara bersamaan, memberikan nilai tambah signifikan bagi Perusahaan [6]. *Chatbot* memberikan pengalaman interaksi yang lebih ramah dan menarik bagi pengguna, dibandingkan dengan pencarian manual melalui daftar pertanyaan yang sering diajukan (FAQ). *Chatbot* menawarkan bantuan yang mudah diakses dan efisien, memungkinkan komunikasi langsung dengan pengguna dan memberikan jawaban yang lebih menarik dan informatif untuk menjawab pertanyaan dan menyelesaikan masalah pengguna [2].

Di era digital ini, banyak organisasi berita memanfaatkan *chatbot* sebagai strategi untuk mempromosikan, menyebarkan, dan merekomendasikan berita kepada *audiens* secara langsung melalui format percakapan. Meskipun penerapan *chatbot* berita telah menunjukkan potensi dalam meningkatkan interaksi dengan *audiens*, efektivitasnya masih beragam dan belum banyak diteliti secara mendalam [7]. Menurut Jones dan Jones, BBC memanfaatkan *chatbot* pada tahun 2019 untuk mendistribusikan berita. Namun, mereka memprediksi bahwa *Character User Interface* (CUI) berbasis teks dan suara akan membawa

perubahan signifikan dalam hubungan antara *audiens* dan penyiar berita di masa depan [8].

Dalam konteks penyampaian informasi, *chatbot* memiliki beberapa keunggulan dibandingkan media tradisional seperti *website* atau aplikasi berita. Keunggulan utama *chatbot* adalah kemampuannya untuk menyampaikan informasi secara *real-time* dan interaktif, sehingga pengguna dapat lebih mudah memahami dan memperoleh informasi yang mereka butuhkan. (*challenges and opportunities with social chatbots*) Seiring perkembangan teknologi, kebutuhan informasi masyarakat semakin tinggi, terutama berita. Kebiasaan mengakses berita melalui internet dan media sosial mendorong pemanfaatan *chatbot* sebagai platform penyedia informasi berita. *Chatbot* mampu menyajikan berbagai jenis berita, mulai dari terkini, lokal, nasional, internasional, hingga olahraga. Keunggulan *chatbot* terletak pada kemampuannya untuk mempersonalisasi informasi berita sesuai dengan minat pengguna [7].

Framework React, sebagai salah satu *framework JavaScript* terpopuler saat ini, menawarkan beberapa keunggulan yang membuatnya ideal untuk pengembangan *chatbot*. Kesederhanaan dan kemudahan penggunaannya menjadikannya pilihan tepat bagi para *developer* dalam membangun *chatbot* yang efektif. Keunggulan ini menjadikan *React* pilihan tepat untuk perancangan *chatbot* penyedia informasi berita berbasis *web*, menawarkan proses pengembangan yang cepat, performa yang tinggi, dan kemudahan personalisasi untuk menghadirkan pengalaman pengguna yang optimal [9].

Meskipun *chatbot* menawarkan berbagai manfaat dalam penyediaan informasi berita, terdapat beberapa tantangan yang perlu diatasi. Salah satu tantangan utama adalah memastikan akurasi dan relevansi informasi berita yang disampaikan. *Chatbot* yang menyediakan informasi tidak akurat atau tidak relevan dapat merusak kepercayaan pengguna dan berdampak negatif pada reputasi *chatbot*. Oleh karena itu, perancangan *chatbot* yang mampu memberikan informasi berita yang akurat dan relevan dengan kebutuhan pengguna menjadi fokus utama dalam perancangan *chatbot* berita

Pemanfaatan *framework React* dalam perancangan *chatbot* penyedia informasi berita berbasis web membuka peluang besar untuk meningkatkan aksesibilitas masyarakat terhadap informasi berita. *Chatbot* menawarkan penyampaian informasi secara *real-time*, interaktif, dan personal, menjadikannya alternatif menarik bagi media tradisional. Penelitian ini bertujuan mengembangkan *chatbot* penyedia informasi berita berbasis web dengan *framework React* yang mampu memberikan informasi berita yang akurat, relevan, dan bermanfaat bagi masyarakat.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, dapat diuraikan rumusan masalah sebagai berikut:

1. Bagaimana rancangan *Chatbot* pada informasi layanan berita berbasis web menggunakan *framework react*?
2. Bagaimanakah efektifitas dari rancangan *Chatbot* informasi layanan berita berbasis web *framework react*?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah disebutkan, tujuan dari penelitian pemanfaatan *Chatbot* penyedia informasi dan berita berbasis web sebagai berikut:

1. Membuat dan mengimplementasikan rancangan *Chatbot* informasi layanan berita berbasis web dengan menggunakan *framework react*.
2. Mengetahui efektifitas dari rancangan *Chatbot* informasi layanan berita berbasis web *framework react*.

1.4 Manfaat Penelitian

Berdasarkan dari rumusan masalah yang sudah penulis sebutkan manfaat dari penelitian pemanfaatan *Chatbot* penyedia informasi dan berita berbasis web sebagai berikut:

1. Menjadikan peningkatan aksesibilitas informasi, *Chatbot* dapat menyediakan akses yang lebih mudah dan cepat bagi pengguna untuk memperoleh informasi berita terkini.
2. Memberikan efisiensi waktu dan biaya, *Chatbot* dapat mengurangi kebutuhan akan sumber daya manusia untuk menjawab pertanyaan umum

terkait berita.

3. Menghasilkan dan mendorong adopsi teknologi baru seperti *chatbot* dan *framework React* dalam industri berita, sehingga membantu meningkatkan inovasi baru pada *chatbot*.

1.5 Batasan Masalah

Dari rumusan masalah yang sudah dibuat sebelumnya maka dapat diuraikan batasan masalahnya sebagai berikut:

1. Peneliti hanya merancang *chatbot* pada informasi berita untuk memudahkan *user* mencari berita berbasis web.
2. Peneliti hanya fokus pada bagian layanan informasi berita.
3. Pemanfaatan *chatbot* pada informasi berita hanya digunakan dalam rangka penelitian dan tugas akhir.

1.6 Sistematika Penulisan

BAB I Pendahuluan:

Bab ini memuat: latar belakang, rumusan masalah, tujuan dan manfaat dan keterbatasan masalah serta sistematika penulisan.

BAB II Kajian Literatur:

Bab ini membahas tentang istilah-istilah teori yang berkaitan dengan penelitian, tahapan penelitian dan kajian-kajian yang berkaitan.

BAB III Metodologi Penelitian:

Bab ini akan menjelaskan proses yang digunakan penulis dalam menyusun penelitian, termasuk tahapan penelitian hingga desain penelitian.

BAB IV Implementasi dan Evaluasi:

Bab ini menjelaskan implementasi dan evaluasi yang terlibat dalam mempelajari analisis, persyaratan, dan desain sistem. Kemudian akan dilakukan pengujian dan mencakup evaluasi terhadap hasil implementasi.

BAB V Kesimpulan dan Saran

Bab ini menjelaskan kesimpulan penelitian tugas akhir serta saran yang mungkin berguna bagi peneliti selanjutnya yang ingin melakukan penelitian dengan topik serupa.

BAB II

KAJIAN LITERATUR

Pada bab ini akan membahas terkait kajian literatur yang akan dijelaskan terkait istilah teori-teori yang berkaitan dengan penelitian secara ringkas, untuk menggambarkan alur penelitian, serta mengetahui keterkaitan penelitian dengan penelitian terkait pada bab sebelumnya

2.1 Website

Website merupakan kumpulan halaman yang memuat informasi dalam bentuk data digital. Informasi ini dapat berupa teks, gambar, animasi, suara, video, atau kombinasi dari semuanya. Melalui penggunaan HTML, CSS, dan *JavaScript*, menjadi pondasi utama dalam membangun struktur dan isi *website*[9]. Hasilnya, informasi pada *website* dapat ditampilkan dalam bentuk yang mudah dibaca dan dinikmati oleh pengguna. Dengan kemudahan akses dan kekayaan informasi yang ditawarkan, *website* telah menjadi bagian tak terpisahkan dari kehidupan *modern*. *Website* menjadi sumber belajar, *platform* bisnis, media komunikasi, dan sarana hiburan bagi individu maupun organisasi di seluruh dunia.

2.2 Chatbot

Chatbot adalah program perangkat lunak komputer yang melakukan percakapan melalui metode pendengaran atau teks. Perangkat lunak ini digunakan untuk melakukan tugas-tugas seperti merespon pengguna dengan cepat, menginformasikan mereka, membantu pembelian produk, dan memberikan layanan yang lebih baik kepada pelanggan. *Chatbot* adalah program yang bekerja pada *platform* Kecerdasan Buatan (AI) & Pembelajaran Mesin (*Machine Learning*). *Chatbot* menjadi semakin populer di kalangan kelompok bisnis saat ini karena dapat mengurangi biaya layanan pelanggan dan menangani banyak pengguna secara bersamaan. Namun, untuk menyelesaikan banyak tugas, diperlukan upaya untuk membuat *chatbot* seefisien mungkin [2].

Ada tiga model klasifikasi yang umum digunakan dalam pengembangan *chatbot*, yaitu *rule-based model*, *retrieval based model*, and *generative model*.

Model-model ini menghasilkan respon yang sesuai berdasarkan metode pemrosesan input dan pembangkitan respon yang diterapkan. Dalam penelitian ini, model yang akan digunakan adalah model berbasis pengambilan (*Retrieval-based model*). Model ini menawarkan fleksibilitas yang lebih tinggi karena memanfaatkan API untuk menanyakan dan menganalisis sumber daya yang tersedia. *Chatbot* berbasis *retrieval based model* mengambil beberapa kandidat respon dari *indeks* dan menerapkan pendekatan pencocokan untuk memilih respon yang paling informatif dan bermanfaat bagi pengguna [4].

2.3 Framework React

React, pustaka *JavaScript open-source* yang populer, memberdayakan developer untuk membangun antarmuka pengguna (UI) web yang canggih. Pendekatan berbasis komponennya memungkinkan pembuatan UI yang kompleks dengan merakit bagian-bagian yang lebih kecil dan dapat digunakan kembali. *Framework React* dikembangkan oleh Meta yang sebelumnya dikenal dengan Facebook dan komunitas developer yang aktif secara konsisten memelihara dan mengembangkan *React*.

Kemampuan *React* melampaui aplikasi *single-page*. Pustaka ini dapat digunakan untuk membangun aplikasi *mobile* dan, dengan *framework* seperti *Next.js*, bahkan aplikasi yang di *render* di *server*. *React* berfokus pada UI dan *rendering* komponen dalam *Document Object Model* (DOM), sehingga aplikasi *React* sering kali membutuhkan pustaka tambahan untuk menangani fungsi seperti *routing* dan interaksi *client-side* lainnya.

Konsep komponen menjadi inti pengembangan *React*. Setiap komponen merupakan blok pembangun UI yang unik, dengan logika dan tampilan visualnya sendiri. Komponen dapat bervariasi dari tombol sederhana hingga seluruh halaman web, memberikan fleksibilitas dan modularitas dalam membangun UI [9].

2.4 API

Application Programming Interface (API) adalah antarmuka yang memungkinkan berbagai aplikasi berkomunikasi satu sama lain. API dapat dikategorikan menjadi dua jenis, yaitu API lokal yang diakses secara langsung dalam aplikasi dan API web yang diakses melalui internet. Penyedia API membangun metode dan fungsi khusus dan menawarkan antarmuka yang mudah digunakan oleh pengembang perangkat lunak (konsumen API) dalam aplikasi mereka. API lokal umumnya diakses secara statis dan terhubung langsung ke aplikasi, sedangkan API web (WAPI) diekspos melalui internet dan diakses melalui URL ke sumber WAPI [10].

Dokumentasi API umumnya dibuat dalam format teks biasa atau mengikuti standar terbuka, namun interaktivitasnya masih minim. Idealnya, dokumentasi API harus disajikan secara interaktif melalui antarmuka web yang memungkinkan pengguna untuk berinteraksi langsung dengan dokumentasi tersebut. Salah satu contoh dokumentasi API interaktif adalah pembuatan dan desain *frontend web* berdasarkan dokumentasi OpenAPI. Namun, alat ini masih membutuhkan sentuhan manual dari pengembang untuk meningkatkan kualitas dokumentasi bagi pengguna akhir. Dokumentasi API yang interaktif dan mudah dipahami bagi pengguna non-teknis masih menjadi tantangan yang perlu diatasi agar API dapat digunakan secara optimal oleh semua kalangan [11].

2.5 NPM

Node Package Manager (NPM) adalah *registry* perangkat lunak terbesar di dunia, dengan lebih dari 2,1 juta paket terdaftar. Dikembangkan khusus untuk *JavaScript*, npm menjadi alat yang esensial bagi para *developer* untuk memanfaatkan paket-paket tersebut dalam membangun aplikasi. NPM umumnya digunakan melalui *Command Line Interface* (CLI), menyediakan antarmuka baris perintah yang mudah digunakan dan *powerful*. Keberadaan npm sebagai *library* dan *registry JavaScript* memungkinkan *developer* untuk menemukan, menginstal, dan mengelola paket-paket yang dibutuhkan dengan mudah dan efisien.

Keunggulan npm sebagai *package manager* terletak pada kemampuannya dalam menangani *dependency tree* (pohon dependensi). npm secara otomatis menginstal dependensi yang dibutuhkan oleh paket, sehingga menyederhanakan pengelolaan dependensi *project* dan meminimalisir konflik antar dependensi. npm menyediakan berbagai macam perintah untuk mengelola paket, menginisialisasi *project* baru, menjalankan skrip, dan mempublikasikan paket ke *registry*. NPM memungkinkan *developer* untuk menentukan skrip kustom dalam file "*package.json*". Skrip ini dapat digunakan untuk berbagai tugas, seperti membangun, menguji, dan menyebarkan aplikasi. *Scripting* ini membantu dalam pemeliharaan dan skalabilitas *project* dengan mengemas *workflow* kompleks menjadi program sederhana yang dapat dieksekusi [12].

2.6 BlackBox

Pengujian *black box*, juga dikenal sebagai pengujian fungsional, merupakan metode pengujian perangkat lunak yang populer karena tidak memerlukan akses ke kode sumber. Pengujian ini berfokus pada pengujian antarmuka sistem, mensimulasikan interaksi pengguna dengan sistem melalui *input* dan *output*. Penguji *black box* tidak memiliki akses ke kode program, sehingga mereka mengandalkan spesifikasi dan dokumentasi API untuk memahami fungsionalitas sistem. Hal ini memungkinkan mereka untuk menguji sistem secara objektif dan independen dari implementasi internalnya. Pengujian *black box* mencakup berbagai skenario input, baik valid maupun tidak valid, untuk mensimulasikan berbagai kondisi penggunaan dan mengidentifikasi potensi kesalahan atau kelemahan [13].

Pengujian *black-box*, metode pengujian yang umum digunakan, bergantung pada spesifikasi API yang terdefinisi dengan baik. *OpenAPI Specification* (OAS) berperan penting dalam hal ini, menyediakan deskripsi API yang detail dan terstruktur. Dengan memanfaatkan OAS, manusia dan komputer dapat memahami kemampuan layanan API tanpa perlu mengakses kode sumber. Keunggulan utama OAS terletak pada kemampuannya memfasilitasi otomatisasi pengujian *black-box*. Spesifikasi OAS yang komprehensif memungkinkan alat pengujian untuk secara otomatis menghasilkan skenario pengujian berdasarkan

definisi API. Skenario pengujian ini mencakup berbagai kombinasi nilai input, baik acak, *default*, maupun data dari sumber eksternal seperti *file* kamus atau respons API sebelumnya [14]. Tujuan utama pengujian *black box* adalah untuk memvalidasi apakah sistem telah memenuhi persyaratan fungsional yang ditentukan dan berfungsi dengan baik sesuai dengan ekspektasi pengguna.

2.7 Penelitian Terkait

Tabel 2. 1 Penelitian Terkait

No	Penulis	Uraian Singkat	Subject	Hasil
1	P D Larasati, A Irawan, S Anwar, M F Mulya, M A Dewi, I Nur Fatima. (2022)	Pada penelitian “Chatbot helpdesk design for digital customer service” berfokus pada pembuatan desain chatbot helpdesk menggunakan WhatsApp yang dimanfaatkan untuk layanan pelanggan digital berdasarkan penjelasan yang diberikan. Chatbot berperan membantu tugas <i>customer service</i> dalam melayani pelanggan. Pelanggan bisa mendapatkan informasi dengan cepat dan menerima solusi untuk semua permasalahan layanan yang mereka alami [5].	Chatbot WhatsApp untuk digital service customer.	Chatbot WhatsApp.
2	Rani Natadian Astuti dan Muhamad Fatchan(2019)	Pada penelitian “PERANCANGAN APLIKASI TEKNOLOGI CHATBOT UNTUK INDUSTRI KOMERSIAL 4.0” Metode yang digunakan dalam penelitian ini yaitu <i>Term Frequency – Inverse Document Frequency</i> (TF-IDF) untuk melakukan perhitungan bobot pada setiap kata dalam dokumen [15].	Chatbot Sebagai Customer Service	Chatbot mampu menangani pertanyaan dengan pola yang berbeda-beda, namun tetap dapat memberikan jawaban yang sesuai.
3	Bima Huberta, Anugrah Bagus Wijaya(2023)	Pada penelitian “PERANCANGAN CHATBOT WEBSITE PROGRAM STUDI INFORMATIKA MENGGUNAKAN FRAMEWORK CODEIGNITER” menggunakan Metodologi <i>waterfall chatbot</i> . Sehingga hasil akhir dari penelitian ini adalah fungsi <i>chatbot</i> yang terintegrasi dengan pengelolaan <i>database chat</i> , sehingga informasi yang diterima oleh pengguna selalu update dengan perkembangan mata kuliah ilmu komputer [16].	Perancangan asisten virtual chatbot pada Universitas Amikom Purwokerto.	asisten virtual chatbot pada Universitas Amikom Purwokerto untuk memudahkan akses mahasiswa terhadap informasi perkuliahan.

4	Ardiansyah(2023)	Pada penelitian “Pendampingan Perancangan <i>Chatbot</i> Sebagai Media Interaktif Dalam Menghadapi Tantangan Era Digitalisasi” peneliti memberikan pendampingan kepada peserta dalam merancang atau membuat <i>chatbot</i> pada platform Smojo.Ai bagi guru, dosen, dan umum sesuai profesi atau kebutuhan peserta dalam bidang tugasnya [17].	Perancangan <i>Chatbot</i> menggunakan <i>platform</i> Smojo.ai	<i>Chatbot</i> untuk guru, dosen, dan umum sesuai dengan kebutuhan masing-masing.
5	Muhammad Rizki Herfian , Ahmad Rio Adriansyah (2021)	Pada penelitian “ANALISIS DAN PERANCANGAN APLIKASI <i>CHATBOT</i> DALAM PELAYANAN PENERIMAAN MAHASISWA BARU PADA PERGURUAN TINGGI” menggunakan <i>Messaging API</i> Telegram dengan menggunakan bahasa pemrograman <i>Python</i> Program diuji dengan menggunakan metode <i>Black Box Testing</i> sedangkan pengujian kepada user menggunakan metode <i>User Acceptance Testing (UAT)</i> dan Kuesioner [18].	<i>Chatbot</i> informasi Penerimaan Mahasiswa Baru (PMB) di STT Nurul Fikri.	<i>Chatbot</i> layanan informasi Penerimaan Mahasiswa Baru (PMB) di STT Nurul Fikri.
6	Rifky Kurniawan, Muhamad Zulfikar	Pada Penelitian “PERANCANGAN APLIKASI <i>CHATBOT</i> PADA SALAH SATU INDUSTRI E-COMMERCE DI JAKARTA” penerapan <i>chatbot</i> dilakukan pada aplikasi mobile. Metode pengumpulan data yang digunakan yaitu studi kasus dengan metode pembuatan secara <i>waterfall</i> , melakukan analysis, design dan implementation. Dengan tujuan penerapan <i>chatbot</i> , didapatkan hasil yang efektif untuk penggunaan <i>chatbot</i> sebagai alternatif dalam respon pesan penjual kepada customer [6].	Perancangan <i>chatbot</i> pada sistem aplikasi <i>e-commerce</i> .	<i>Chatbot</i> Sebagai layanan <i>customer</i> otomatis.

STT - NF

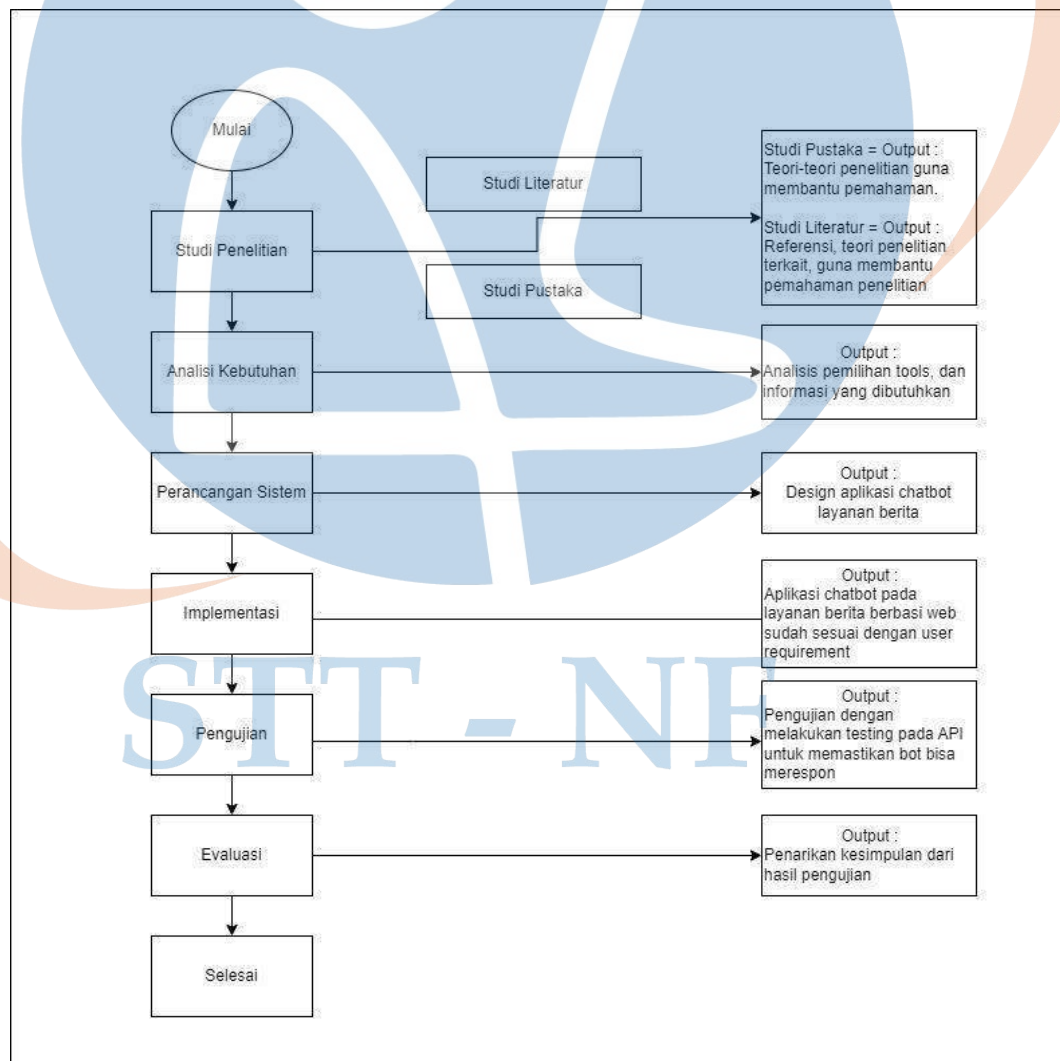
BAB III

METODOLOGI PENELITIAN

Pada bab ini dijelaskan terkait alur yang diuraikan penulis dalam penelitian, diantaranya terdapat tahapan penelitian serta rancangan penelitian. Tahapan penelitian akan menjelaskan bagaimana proses dari penulis lakukan pada penelitian ini dari awal sampai akhir penelitian.

3.1 Tahapan Penelitian

Tahapan yang dilakukan dalam perancangan aplikasi *chatbot* pada layanan berita berbasis *web* menggunakan *framework react* secara umum, akan dijelaskan pada gambar berikut :



Gambar 3.1 Tahapan Penelitian

3.1.1 Studi Penelitian

Fase ini meliputi pengumpulan informasi dan pemahaman awal terhadap topik penelitian, pencarian literatur, pencarian referensi terhadap jurnal penelitian yang relevan untuk memudahkan pemahaman terhadap penelitian, serta observasi dan penelitian untuk memahami subjek penelitian.

3.1.2 Analisis Kebutuhan

Tahap analisis kebutuhan mengidentifikasi dan memahami kebutuhan dan masalah yang ingin dipecahkan oleh penelitian ini. Identifikasi tujuan penelitian utama, kebutuhan, dan hambatan yang mungkin terjadi saat menerapkan solusi. Analisis ini membantu dalam merumuskan strategi dan pendekatan yang tepat untuk memecahkan masalah yang diteliti.

3.1.3 Perancangan Sistem

Proses perancangan *chatbot* pada layanan berita berbasis web dilakukan sesuai dengan kebutuhan yang ada, yaitu pencarian berita, pemilihan kategori berita, dan pengetikan *keyword* yang sesuai *search*. Dengan adanya *chatbot* pada layanan berita ini diharapkan *user* bisa dengan mudah mendapatkan berita dan memudahkan pengguna untuk melakukan *search* berita yang diinginkan.

3.1.4 Implementasi

Selanjutnya setelah membuat desain antarmuka *chatbot* pada layanan berita berbasis web yang telah sesuai dengan kebutuhan yang ada. Kemudian dilakukan implementasi terhadap perancangan *chatbot* pada layanan berita berbasis web dengan menggunakan *framework react*.

3.1.5 Pengujian

Setelah implementasi, tahap selanjutnya adalah pengujian sistem dilakukan untuk memastikan bahwa *chatbot* berfungsi sebagaimana mestinya. Ini mencakup pengujian fungsionalitas, keandalan, kinerja, dan *hit* API.

3.1.6 Evaluasi Hasil

Tahap terakhir yaitu melakukan evaluasi terhadap aplikasi *chatbot* yang telah selesai dirancang dan diimplementasikan. Dari tahapan evaluasi ini mendapatkan hasil dari pengujian yang dilakukan oleh *user*, sehingga dapat mengetahui apakah aplikasi *chatbot* pada layanan berita menggunakan *framework react* sudah dapat diakses dan mudah untuk digunakan.

3.2 Rancangan Penelitian

3.2.1 Jenis Penelitian

Penelitian ini menggunakan metode R&D (*Research and Development*) karena memiliki karakteristik adanya produk yang dihasilkan dari penelitian ini yaitu berupa *Chatbot*.

3.2.2 Metode Analisis

Metode analisis yang digunakan pada penelitian ini adalah metode deskriptif. Analisis dilakukan dengan mendeskripsikan dan menggambarkan hasil yang diperoleh, untuk mengetahui bagaimana efektivitas dari implementasi *Chatbot* penyedia informasi berita berbasis *web*.

3.2.3 Metode Pengumpulan Data

a. Observasi

Penulis melakukan analisis terhadap masalah yang ada dengan cara studi kasus dan referensi terhadap berita-berita, dan mengumpulkan data yang berhubungan dengan penelitian penulis.

b. Studi Pustaka

Studi pustaka dilakukan untuk mengumpulkan data yang akan digunakan agar mempermudah proses perancangan dengan cara mempelajari serta memahami sumber pustaka yang memiliki hubungan dengan penelitian yang kemudian dijadikan sebagai daftar referensi yang penulis gunakan.

3.2.4 Waktu Penelitian

Tabel 3. 1 Waktu Penelitian

No	Nama Kegiatan	Bulan				
		Feb	Mar	April	Mei	Juni
1	Pengajuan Proposal					
2	Kajian Literatur					
3	Pengumpulan Data					
4	Analisis dan Perancangan					
5	Implementasi Rancangan					
6	Pengujian Aplikasi					
7	Penyusunan Laporan					

3.2.5 Lingkungan Pengembangan

Dalam proses perancangan penulis mengikuti langkah-langkah seperti setup lingkungan, instalasi dependencies, penulisan *code react* dan integrasi dengan API berita *online*, pengujian dan *deployment*.

Bahan dan Alat

a. Perangkat Lunak

1. *Google Chrome* dengan versi 119.0.6045.124 digunakan untuk melihat hasil dari implementasi.
2. *Visual Studio Code* digunakan untuk penulisan *code*.
3. Postman versi digunakan sebagai membangun, menguji dan modifikasi API.
4. Sistem Operasi yang digunakan adalah Windows 11 PRO 64bit.

b. Perangkat Keras

Perangkat yang digunakan dalam implementasi *chatbot* penyedia informasi berita berbasis *web* adalah laptop dengan spesifikasi sebagai berikut:

Tabel 3. 2 Spesifikasi Perangkat Keras

1. Model	LENOVO 21A2
2. Processor	AMD Ryzen 3 5300U
3. RAM	12 GB
4. SSD	256 GB

3.2.6 Metode Pengujian

Metode pengujian dalam penelitian pengembangan *chatbot* pada layanan berita berbasis web menggunakan *framework react* adalah dengan menggunakan metode *Black box*, yaitu pengujian yang berfokus pada persyaratan fungsional perangkat lunak untuk melihat apakah memberikan hasil yang diinginkan dan sesuai fungsionalitas dari sistem yang telah dibuat, pengujian fungsional dilakukan oleh pengembang, berikut adalah beberapa yang nantinya penulis lakukan dalam pengujian aplikasi yang sudah dirancang :

Tabel 3. 3 Template Tabel Pengujian

No	Kategori Untuk Pengujian	Keterangan	Hasil
1.	<i>User</i> berhasil <i>Login</i>	Pada pengetesan di beberapa akun yaitu <i>user</i> bisa dan berhasil <i>login</i>	Lolos/Tidak Lolos
2.	<i>User</i> dapat berinteraksi dengan Bot	Dengan demikian <i>user</i> dapat berinteraksi dengan bot pada halaman	Lolos/Tidak Lolos
3.	<i>User</i> dapat memilih kategori berita	Serta <i>user</i> dapat memilih kategori berita yang diinginkan dan sudah ada di	Lolos/Tidak Lolos
4.	<i>User</i> dapat memilih berita yang disediakan oleh Bot	Lebih lanjut <i>user</i> bisa memilih berita dengan kategori yang tertera dan berhasil	Lolos/Tidak Lolos
5.	Bot berhasil mendeteksi <i>keyword</i> yang tidak ada dengan balikan auto <i>response error</i>	Bot tidak bisa menjawab <i>keyword</i> yang tidak ada dengan <i>respons error</i>	Lolos/Tidak Lolos

Peneliti menggunakan tabel dalam kerangka untuk pengujian guna memastikan perancangan *chatbot* semua berjalan dan sukses dan sesuai fungsional nantinya.

Template tabel pengujian diatas memungkinkan peneliti untuk mencatat apakah setiap fitur berfungsi sebagaimana mestinya (Lolos) atau tidak (Tidak Lolos). Ini adalah tabel yang berguna untuk memastikan bahwa semua fungsi utama sistem bekerja dengan baik nantinya

BAB IV

IMPLEMENTASI DAN EVALUASI

Pada bab ini menjelaskan tentang implementasi antar muka berupa aplikasi *chatbot* pada layanan berita berbasis web dan membahas hasil dari pengujian serta evaluasi dari aplikasi.

4.1. Analisis dan Perancangan

Tahapan analisis dan perancangan merupakan tahapan penting dalam pengembangan atau perancangan sistem pada perancangan *chatbot* informasi layanan berita berbasis web, sehingga analisis dan perancangan sangat dibutuhkan dan saling bekerja sama untuk memastikan bahwa solusi yang dirancang atau dikembangkan tidak hanya memenuhi kebutuhan pengguna, namun juga bisa diterapkan secara efektif dan efisien.

Dalam bagian ini, akan diuraikan secara komprehensif mengenai metodologi penelitian yang diadopsi, dengan penekanan khusus pada tahapan analisis sistem yang mendalam serta perancangan sistem yang terstruktur yang menjadi pondasi dari penelitian ini.

4.1.1 Analisis Sistem

Sejalan perkembangan teknologi saat ini *chatbot* sangat penting untuk membantu menyelesaikan dan memudahkan pekerjaan dan kebutuhan pada aktivitas sehari-hari. Namun masih banyak juga informasi-informasi seperti halnya berita yang belum ada layanan *chatbot* untuk mencari berita yang relevan dan sesuai, maka dari itu perancang memberikan kemudahan dengan memberikan layanan *chatbot* informasi berita.

Chatbot adalah program komputer yang dirancang untuk berinteraksi dengan pengguna melalui antarmuka percakapan berbasis teks atau pesan suara menggunakan sistem respons teratur. Sistem ini didasarkan pada seperangkat aturan dan perintah yang telah ditetapkan pada bab sebelumnya yang memungkinkan *chatbot* mengenali *keyword* tertentu dan memberikan respon yang sesuai melalui *keyword*.

Untuk memastikan *chatbot* bisa berinteraksi ada *chatbot* informasi layanan berita berbasis web. Berdasarkan analisis sistem yang dilakukan pada perancangan *chatbot* pada layanan informasi berita berbasis web dengan menggunakan *Framework React*, dapat disimpulkan beberapa poin penting sebagai berikut:

1. Implementasi chatbot pada layanan informasi berita berbasis web dengan menggunakan *framework react* memerlukan berbagai aspek teknologi seperti, API, pemrosesan serta pengalaman pengguna.
2. Sistem *chatbot* yang dirancang akan menghasilkan suatu sistem yang mempermudah pengguna dalam kegiatan pencarian berita dengan cepat dan terbaru.
3. Sistem dari *chatbot* layanan berita ini pengguna akan mendapatkan berita informatif yang terkait dalam setiap harinya.

4.1.2 Kebutuhan Sistem

Pada kebutuhan sistem perancangan *chatbot* informasi pada layanan berita berbasis web diperlukan beberapa kebutuhan untuk membuat membuat sistem dari *chatbot* itu sendiri, berikut adalah beberapa kebutuhan *hardware*, *software*, serta yang dibutuhkan :

1. Perangkat keras

Perangkat keras merupakan komponen fisik dari sebuah sistem dalam perancangan *chatbot* untuk membantu perancangan *chatbot* dan keberhasilan perancangan dari *chatbot*. Pada pembuatan *chatbot* informasi layanan berita berbasis web perangkat keras yang digunakan oleh perancang sebagai berikut :

Tabel 4.1 Spesifikasi Perangkat Keras

Spesifikasi Perangkat Keras	Spesifikasi
Nama Sistem	LAPTOP-PFVEN7N3
System Manufacturer	LENOVO
Ram	12.0 GB
Tipe Sistem	x64-based PC
keyboard	Standart
VGA	AMD Radeon(TM) Graphics
Version	23H2 (OS Build 22631.3593)
DirectX Version	DirectX 12
SSD	256GB SSD M.2 2242 PCIe 3.0×4 NVMe
Operasi Sistem	Windows 11 Home Single Language 64-bit (10.0, Build 22631)

2. Perangkat Lunak

Perangkat lunak yang digunakan dalam melakukan perancangan *chatbot* informasi layanan berita berbasis web seperti berikut :

1. VsCode
2. Postman
3. Google Chrome
4. NPM
5. Node.js
6. Draw.io

3. Fitur

Pada fitur ini peneliti membuat sistem *chatbot* yang akan menampilkan berita dengan sesuai apa yang dicari oleh *user*, dengan logika dari pencarian *query* data yang sudah terstruktur.

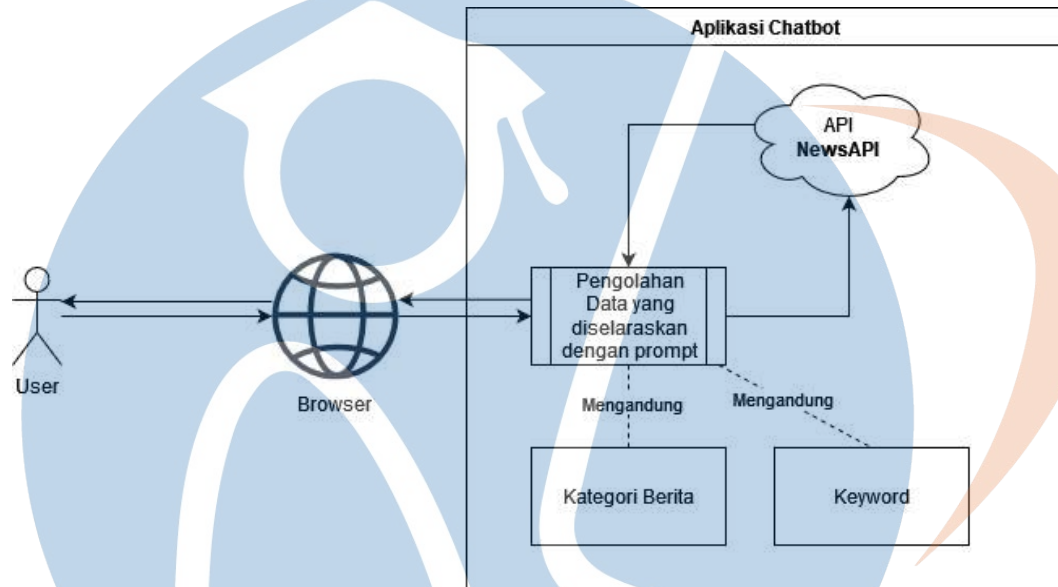
3.6 Menampilkan informasi berita terkini.

3.7 Hasil dari pencarian akan menampilkan sesuai dengan kategori yang dipilih.

3.8 Pada pengembangan layanan informasi layanan *chatbot* ini masih terbuka untuk umum karena perlu penelitian lebih lanjut.

4.1.3 Perancangan sistem

Pada tahapan perancangan sistem ini peneliti memanfaatkan dari sistem informasi layanan *chatbot* berita, yang berisi terdiri dari gambar berikut :



Gambar 4.1 Perancangan Sistem

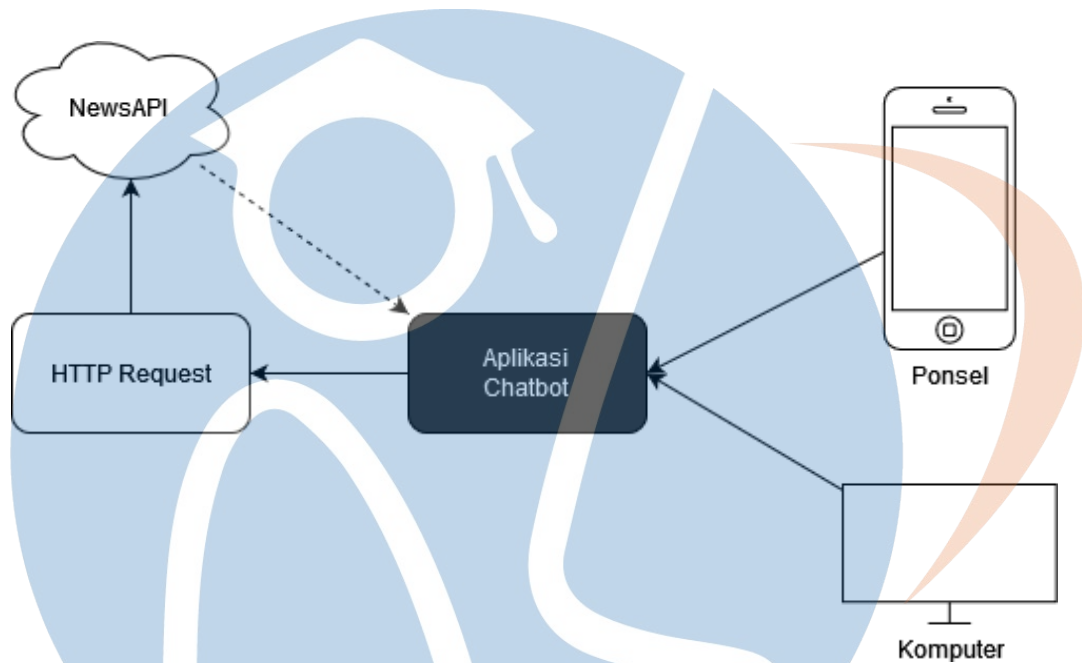
Berdasarkan rancangan pada gambar 4.1, penulis membuat prototipe sederhana untuk mensimulasikan interaksi antara *user*, bot dan API dari pihak ketiga. Interaksi yang terjadi pada proses gambar di atas user mengakses aplikasi *chatbot* didalamnya akan ada interaksi antara *user* dengan bot, *user* akan memilih kategori yang sudah disediakan oleh bot atau *user* bisa memilih kategori/*keyword* secara bebas yang nantinya mengirim *prompt* ke bot untuk mengambil data berita dari *newsAPI*.

Selanjutnya pada tahapan berikutnya peneliti membuat Arsitektur sistem, *Use Case Diagram*, *Activity diagram* serta Rancangan UI (*User Interface*) yang akan digunakan pada tahapan analisis sistem untuk menentukan siapa saja pengguna dan apa saja yang dibutuhkan dalam pembuatan aplikasi layanan berita berbasis web dengan menggunakan *framework react*.

4.1.4 Arsitektur Sistem

Arsitektur sistem merupakan suatu konsep kerja yang mendeskripsikan kerangka dalam perancangan *chatbot* dengan mendeskripsikan bentuk serta tahapan komponen yang saling sesuai satu dengan yang lainnya.

Berikut adalah gambar arsitektur sistem pada pembuatan *chatbot* informasi layanan berita :

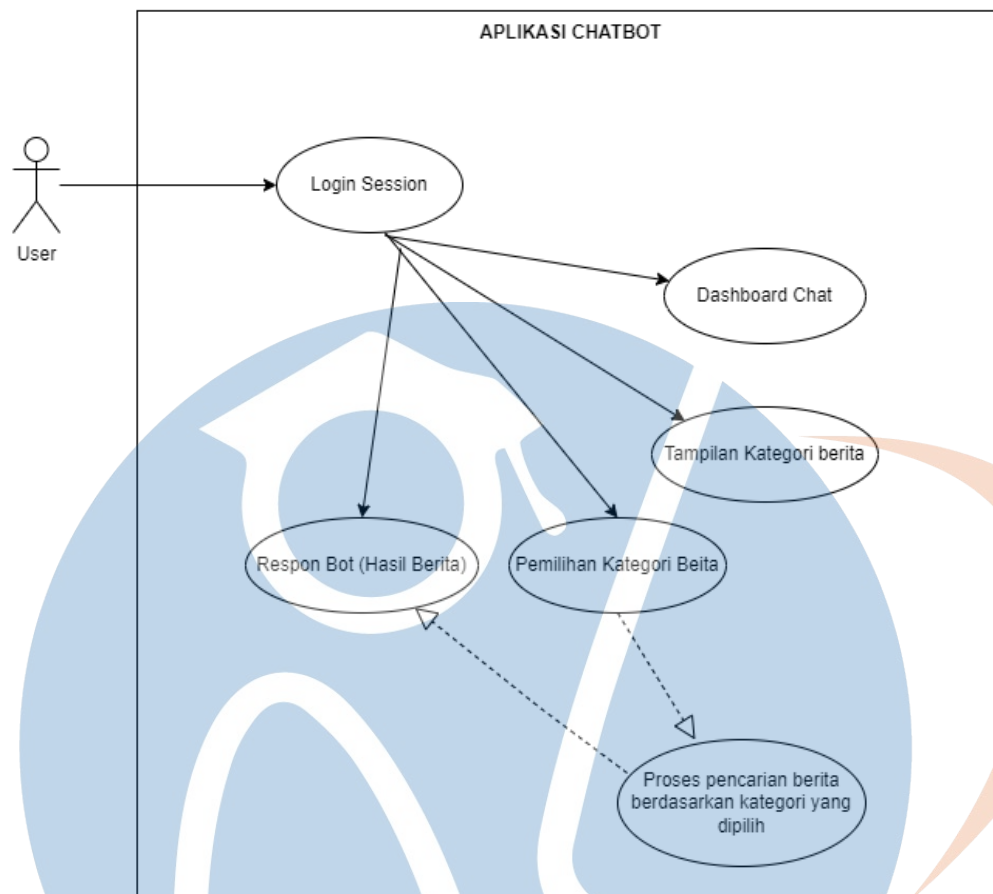


Gambar 4.2 Arsitektur Sistem

Pada gambar 4.2 menjelaskan bahwa ponsel dan komputer berperan penting untuk menampilkan antarmuka pada sistem aplikasi *chatbot* dan juga berperan sebagai alat komunikasi antara *user* dengan *chatbot*, pada aplikasi *chatbot* nantinya *user* akan disuguhkan beberapa kategori yang bisa dipilih oleh *user*. Saat *user* memilih kategori maka aplikasi *chatbot* akan mengirimkan data melalui http request untuk mendapatkan daftar berita sesuai kategori yang dipilih.

4.1.5 Use Case Diagram

Untuk memahami lebih lanjut bagaimana pengguna berinteraksi dengan *chatbot* pada layanan berita berbasis web yang dikembangkan menggunakan framework *react*, mari kita telaah *Diagram Use Case* yang disajikan pada gambar berikut:



Gambar 4. 3 Use Case Diagram

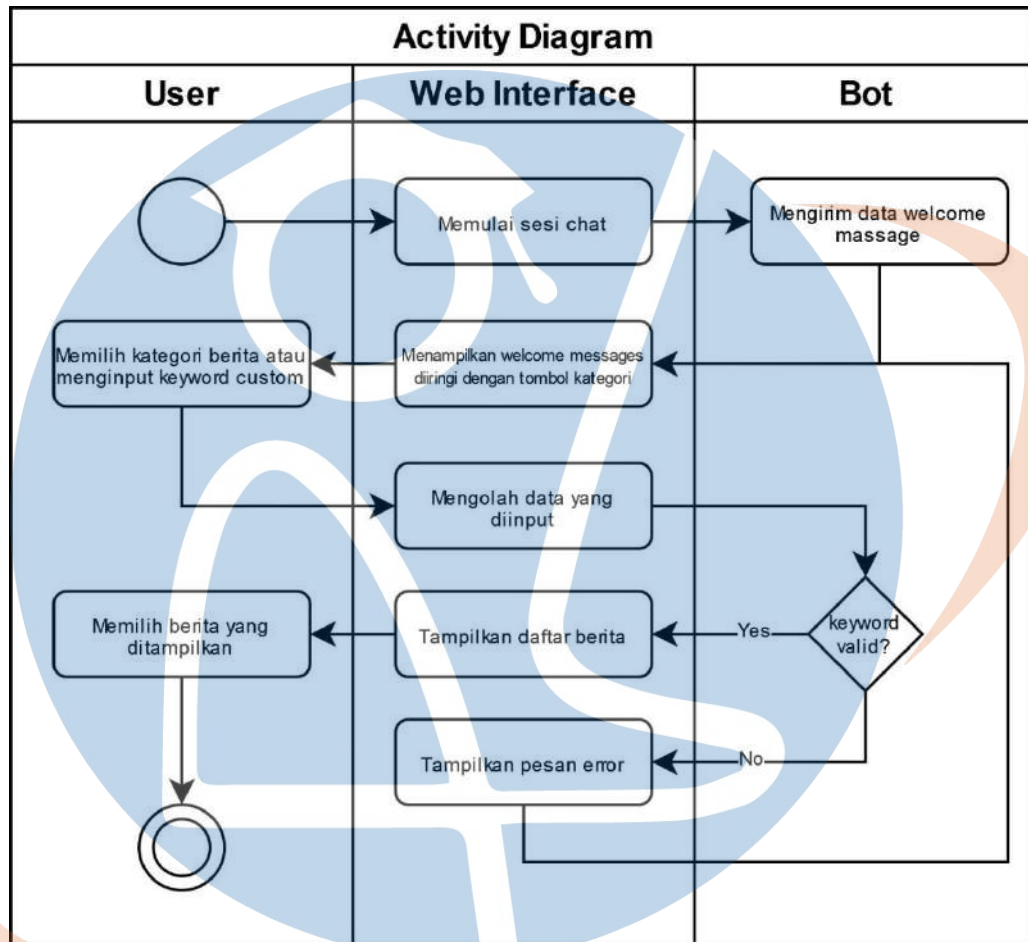
Pada gambar 4.3 diatas dijelaskan terkait *use case* dari *Chatbot* Berita dirancang untuk memberikan pengalaman pengguna yang interaktif dan personal dalam mengakses informasi berita. *Use case* pada gambar diatas juga menggambarkan fungsi utama dan alur interaksi pengguna dengan sistem. Proses dimulai dengan *login*, setelah berhasil masuk, *user* memiliki akses ke fitur pemilihan kategori berita. Selanjutnya dengan *login user* dapat masuk pada tampilan chat utama, memungkinkan interaksi langsung antara *user* dan *chatbot* untuk mencari informasi berita.

Pada gambar *use case* juga menawarkan fitur kategorisasi berita. *User* bisa memilih daftar kategori berita yang tersedia, memberikan gambaran luas tentang cakupan konten. Lebih lanjut, *user* juga memiliki opsi untuk memilih kategori berita sesuai yang diinginkan. Pemilihan kategori berita melalui proses pencarian otomatis. Sistem akan mengumpulkan dan menyaring berita berdasarkan kategori yang dipilih. Hasil pencarian ini kemudian dapat diakses melalui fitur "Melihat hasil berita", memungkinkan *user* untuk mencari berita-berita yang sesuai dengan

yang direkomendasikan oleh bot.

4.1.6 Activity Diagram

Activity diagram menggambarkan sebuah *Workflow* dari sebuah sistem atau proses bisnis yang ada pada perangkat lunak.



Gambar 4. 4 Activity Diagram

Pada gambar 4.4 *activity diagram* ini menggambarkan proses interaksi antara *user*, antarmuka web, dan bot dalam sebuah sistem penyampaian berita berbasis chat. *User* memulai sesi chat yang direspons oleh bot dengan mengirimkan pesan sambutan melalui antarmuka web. *User* kemudian memilih kategori berita atau memasukkan kata kunci kustom. Data yang diinput ini diproses oleh bot untuk memeriksa validitas kata kunci. Jika kata kunci valid, bot menampilkan daftar berita yang sesuai melalui antarmuka web, yang kemudian dapat dipilih oleh *user*. Jika kata kunci tidak valid, bot mengirimkan pesan error yang ditampilkan oleh antarmuka web, meminta *user* untuk mencoba lagi.

Diagram ini mengilustrasikan alur komunikasi dan pengambilan keputusan dalam sistem penyampaian berita berbasis chat.

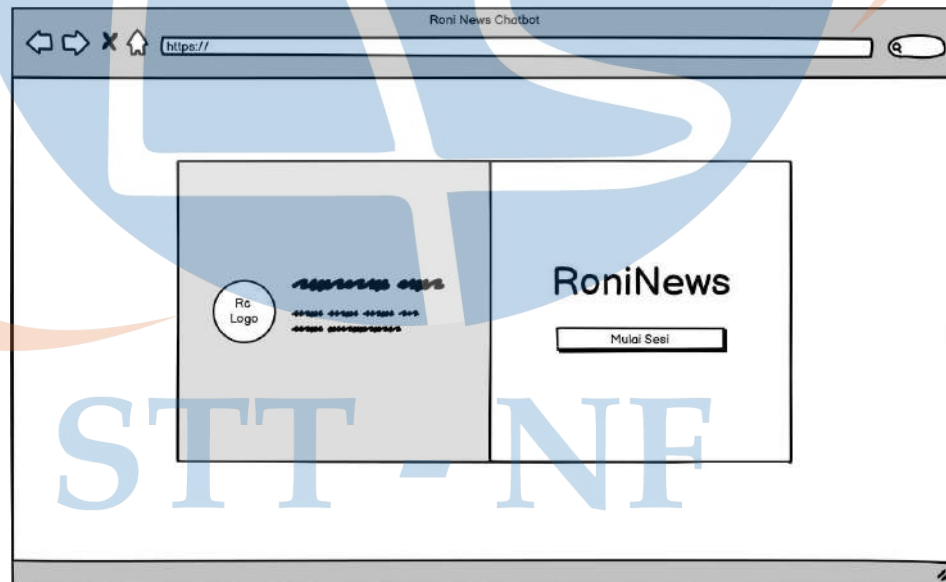
4.1.7 Perancangan UI

Rancangan UI (*User Interface*) menggambarkan alur interaksi pengguna yang sederhana dan langsung, dimulai dari proses antar muka *login* hingga tampilan dengan tujuan memberikan pengalaman yang intuitif dan efisien dalam mengakses dan berinteraksi dengan konten berita melalui antarmuka berbasis *chat*.

Berikut adalah gambar *wireframe chatbot* pada layanan informasi berita berbasis web :

1. *Wireframe login*

Wireframe chatbot pada layanan informasi berita berbasis web dengan menggunakan *framework react* ini menampilkan antarmuka *login* untuk aplikasi "*Roni News Chatbot*", dengan memperlihatkan serta menerapkan prinsip-prinsip desain interaksi dan arsitektur informasi yang efektif.

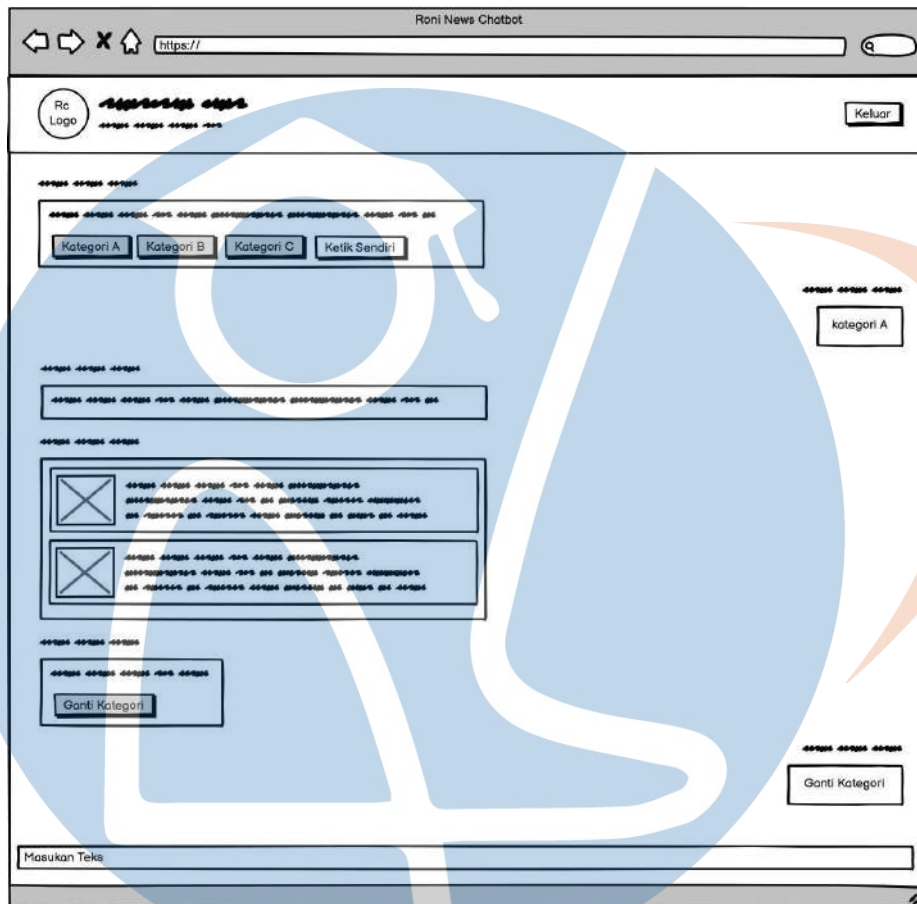


Gambar 4.5 *Wireframe Login*

2. *Wireframe chat*

Pada *Wireframe chat* akan menampilkan halaman utama dari *chatbot* yang menerapkan prinsip desain interaksi serta arsitektur informasi dari berita, dengan bagian bawah dengan bagian bawah antarmuka

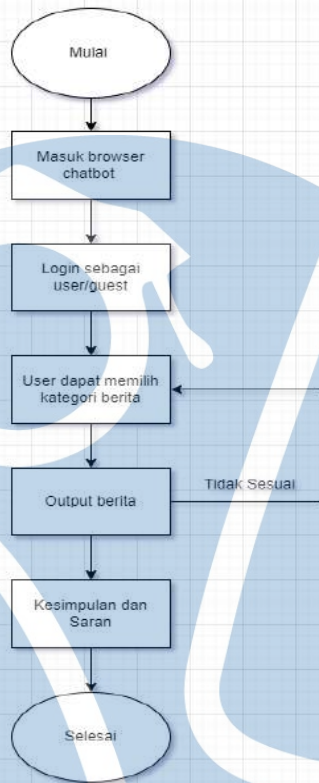
menampilkan kolom input teks, mengonfirmasi munculnya interaksi aplikasi *chatbot*. Penempatan strategis tombol "Ganti Kategori" untuk memudahkan *user* mencari berita dengan dengan kategori yang diinginkan seperti gambar dibawah ini :



Gambar 4. 6 Wireframe Chat

4.1.9 Perancangan Pengujian

Pada tahapan perancangan pengujian, peneliti membuat perancangan *chatbot* pada informasi berita berbasis web menggunakan *framework react*, dimana pada perancangan pengujian ini bertujuan untuk mengetahui sistem pengujian yang dilakukan, sehingga peneliti dapat mengetahui sejauh mana *chatbot* bisa berjalan dan berfungsi. Peneliti membuat gambar perancangan pengujian seperti pada gambar dibawah ini:

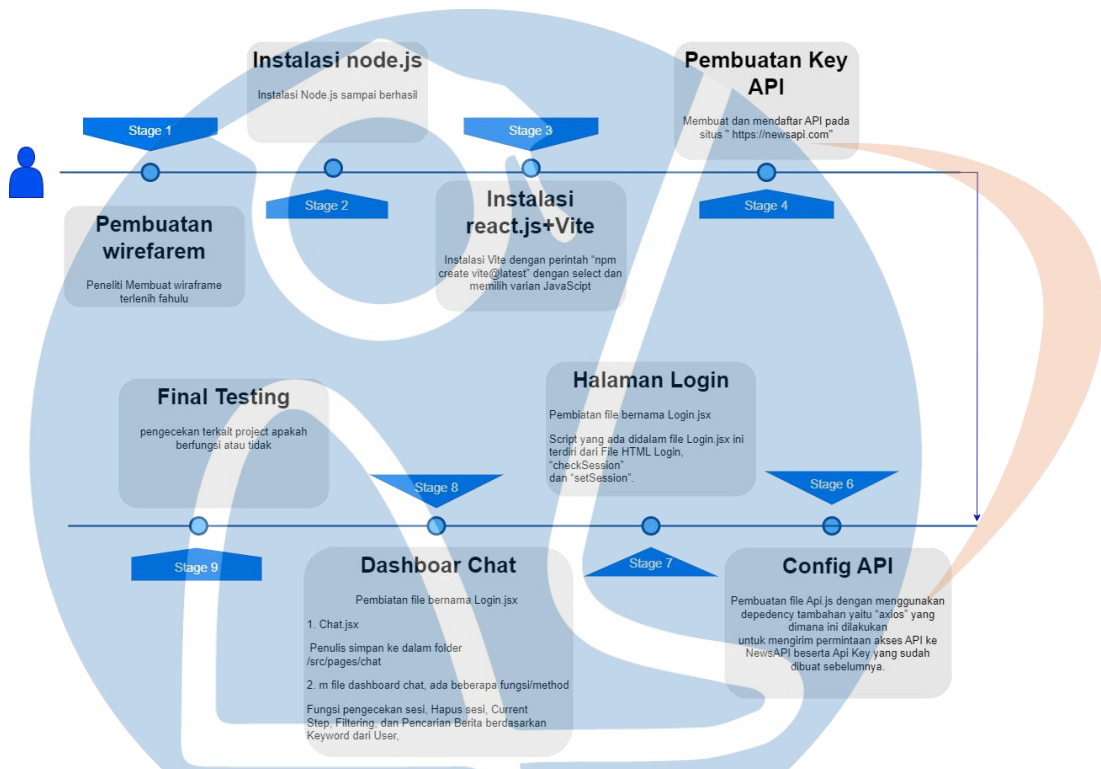


Gambar 4. 7 Perancangan Pengujian

Pada gambar 4.7 di atas merupakan dari sebuah tahapan pengujian dalam perancangan *chatbot* pada layanan informasi berita berbasis web dengan menggunakan *framework react*. Dimana aplikasi *chatbot* diakses melalui web *browser*, kemudian *login* sebagai *user* dan dilakukan pemilihan kategori berita untuk mendapatkan hasil berita. Jika tidak sesuai maka *user* dapat melakukan pemilihan kategori berita kembali.

4.2 Implementasi Sistem

Pada tahapan ini peneliti melakukan perancangan implementasi sistem untuk melakukan perancangan terhadap *chatbot* yang akan dikembangkan dengan tujuan memudahkan peneliti dalam melakukan perancangan pada *chatbot* informasi berita berbasis web menggunakan *framework react* nantinya.



Gambar 4. 8 Implementasi Sistem

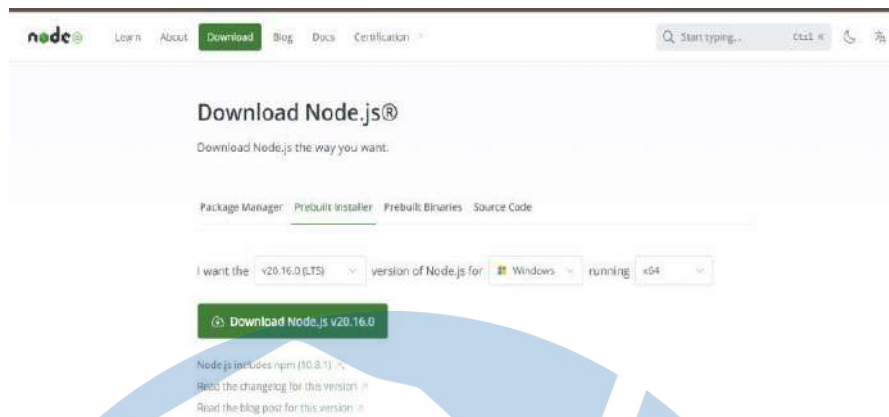
Pada gambar 4.8 diatas dijelaskan terkait implementasi sistem pada pengembangan *chatbot* informasi berita berbasis web menggunakan *framework react*.

Berikut panduan langkah-langkah mengimplementasi awal untuk mengembangkan *Chatbot* dari React JS :

1. Instalasi Node.js di Windows

- Pada tahapan awal dalam merancang chatbot, penulis melakukan instalasi Node.js pada situs resmi Node.js melalui link berikut:

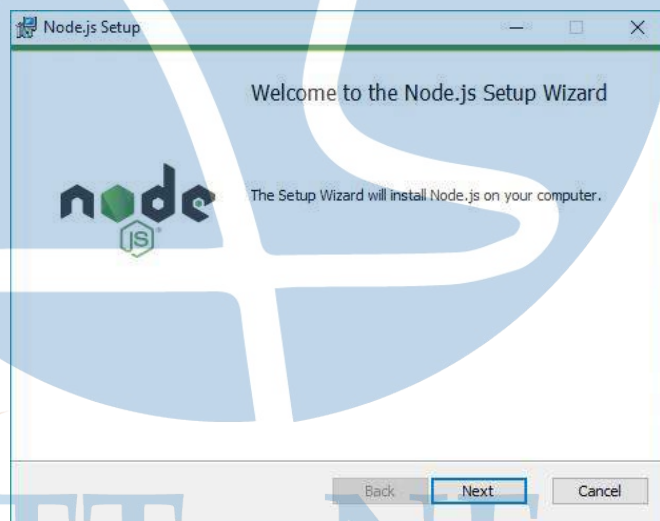
<https://nodejs.org/en/download/prebuilt-installer>



Gambar 4. 9 Instalasi Node.js

Pada gambar diatas penulis melakukan tab "Prebuilt Installer" dan situs web akan secara otomatis mendeteksi versi yang sesuai dengan sistem operasi yang digunakan oleh penulis, dengan melanjutkan instalasi "Download" berwarna hijau pada gambar diatas

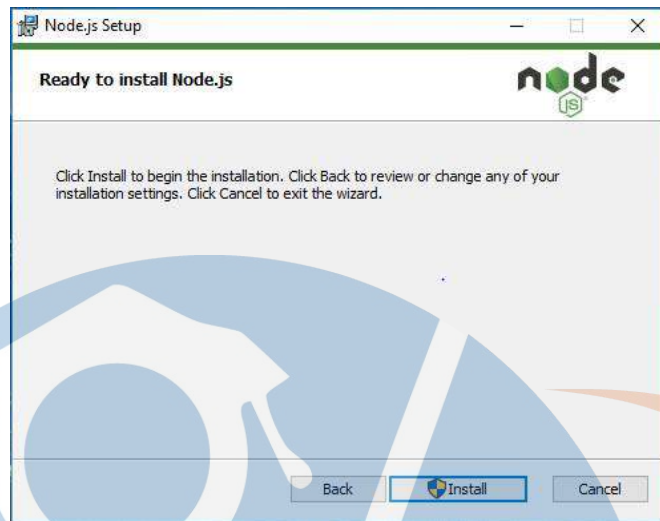
b. Installer Node.js



Gambar 4. 10 Installer Node.js

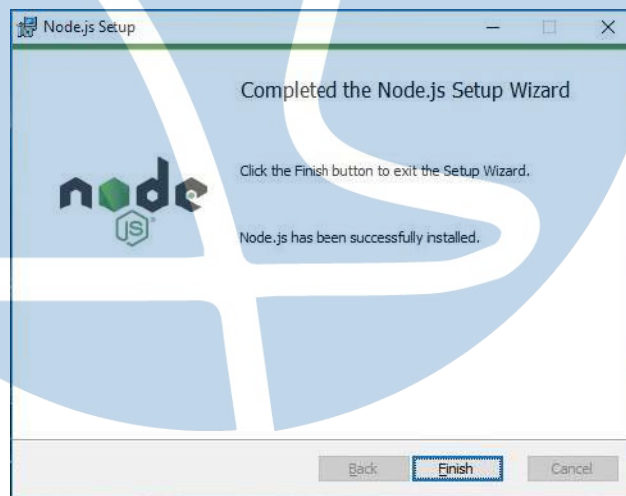
Selanjutnya penulis melakukan proses instalasi dengan mengklik tombol "Next" pada setiap langkah, hingga mencapai tahap akhir yang menampilkan layar berjudul "Ready to Install Node.js". Pada layar ini, akan melihat ringkasan dari semua pilihan yang telah dibuat selama proses instalasi, serta opsi untuk memulai proses pemasangan perangkat lunak Node.js.

c. Install



Gambar 4. 11 Tampilan Memilih Install

d. Instalasi *Node.js* selesai



Gambar 4. 12 Instalasi *Node.js* Selesai

Tampilan pada gambar 4.12 seperti gambar diatas, maka instalasi *Node.js* sukses dan penulis melanjutkan ke langkah berikutnya.

2. Instalasi *React JS + Vite*

Pada tahap ini penulis melakukan eksekusi pada tahap perancangan *chatbot* dengan langkah dibawah ini :

- a. Penulis melakukan instalasi *Vite* dengan perintah “*npm create vite@latest*” dan melanjutkan langkah selanjutnya seperti dibawah ini:

```
Terminate batch job (Y/N)? y
PS C:\Users\ANISA OKTAPIANI\OneDrive\Documents\Roni Prawijaya\Persentasi TA\vite> npm create vite@latest
Need to install the following packages:
create-vite@5.4.0
Ok to proceed? (y)

> chatbot-news@0.0.0 npx
> create-vite

✓ Project name: ... chatbot-news
✓ Select a framework: » React
✓ Select a variant: » JavaScript
```

Gambar 4. 13 Instalasi React JS+Vite

Pada bagian *select a framework*, penulis memilih *React* dan serta pemilihan pada varian yaitu *variant Javascript*

b. Instalasi vite :

```
Scaffolding project in C:\Users\ANISA OKTAPIANI\OneDrive\Documents\Roni Prawijaya\Persentasi TA\vite\chatbot-news...
Done. Now run:

cd chatbot-news
npm install
npm run dev

npm run dev

PS C:\Users\ANISA OKTAPIANI\OneDrive\Documents\Roni Prawijaya\Persentasi TA\vite> cd .\chatbot-news\
PS C:\Users\ANISA OKTAPIANI\OneDrive\Documents\Roni Prawijaya\Persentasi TA\vite\chatbot-news> npm install
npm warn deprecated inflight@1.0.6: This module is not supported, and leaks memory. Do not use it. Check out lru-cache if you want a good and test
ed way to coalesce async requests by a key value, which is much more comprehensive and powerful.
npm warn deprecated @humanwhododes/config-array@0.11.14: Use @eslint/config-array instead
npm warn deprecated rimraf@3.0.2: Rimraf versions prior to v4 are no longer supported
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer supported
npm warn deprecated @humanwhododes/object-schema@2.0.3: Use @eslint/object-schema instead
```

Gambar 4. 14 Instalasi Vite Selesai

Pada gambar 4.14 adalah proses inialisasi proyek web menggunakan *framework Vite* dengan *Node.js* dan *npm*. proses pada gambar tersebut disiapkan melalui tahap *scaffolding*, di mana struktur dasar proyek dibentuk dan ditempatkan dalam *direktori* tertentu. Selanjutnya, perintah ``npm install`` dijalankan untuk menginstal semua dependensi yang diperlukan oleh proyek dan perintah ``npm run dev`` digunakan untuk memulai *server* pengembangan lokal.

c. Perintah “*npm run dev*” untuk *project vite* berjalan

```
found 0 vulnerabilities
PS C:\Users\VANISA OKTAPRIANI\OneDrive\Documents\Roni Prarajaya\Persentasi TA\vite\chatbot-news> npm run dev

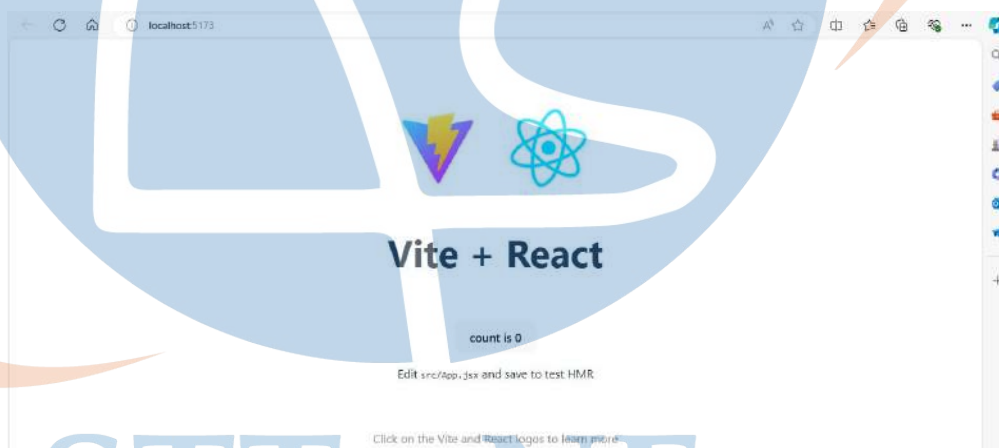
> chatbot-news@0.0.0 dev
> vite

VITE v5.3.5 ready in 864 ms
  → Local:   http://localhost:5173/
  → Network: use --host to expose
  → press h + enter to show help
```

Gambar 4. 15 Jalankan *npm run dev*

Pada gambar diatas menunjukkan terminal yang sedang menjalankan perintah untuk memulai server pengembangan menggunakan *Vite*. Setelah menjalankan perintah “*npm run dev*” *Vite* versi 5.3.5 berhasil diinisialisasi dan siap dalam 864 milidetik. Server ini berjalan pada alamat lokal <http://localhost:5173/>. Pesan tambahan menginformasikan bahwa opsi `--host` dapat digunakan untuk mengakses server dari perangkat lain dalam jaringan yang sama, dan bantuan lebih lanjut dapat diakses dengan menekan tombol "h".

d. Instalasi *React.js* + *Vite* berhasil



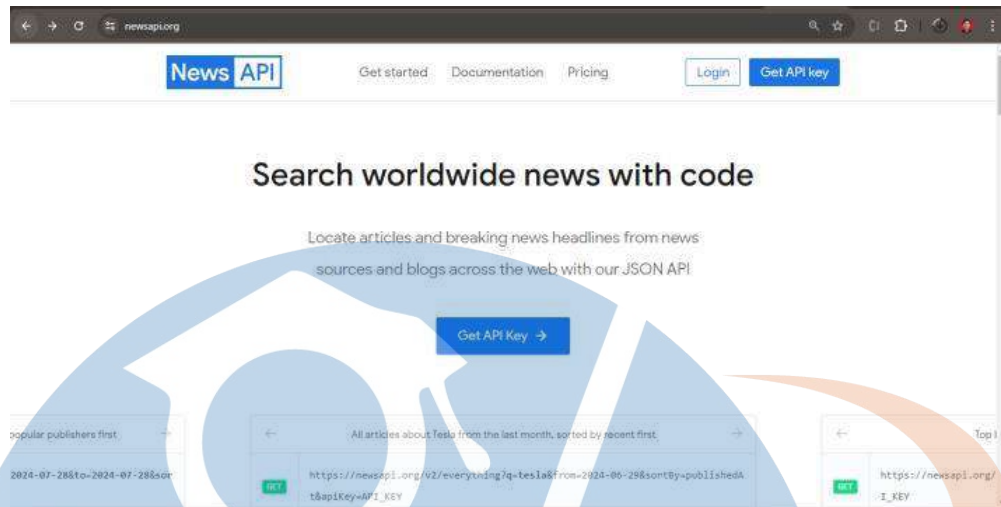
Gambar 4. 16 Instalasi *React.js* + *Vite* Berhasil

Pada gambar 4.16 diatas menjelaskan bahwa fitur dasar dari pengembangan *chatbot* yang akan dirancang dalam instalasi dan konfigurasi proyek berbasis *Vite* dan *React* telah berhasil

3. API dari *NewsAPI*

Pada pembuatan key API penulis mengambil API dari *NewsAPI* dengan langkah-langkah dibawah ini:

- a. Mengunjungi situs <https://newsapi.com>



Gambar 4. 17 Pembuatan Key API

- b. Pemilihan pada “Get API Key” penulis melakukan pendaftaran API Key

Register for API key

First name

Roni Prawjaya

Email address

ronipraw@gmail.com

Email addresses will be verified, please enter a real one. Disposable addresses have been blocked.

Choose a password

You are...

I am an individual

I am a business, or am working on behalf of a business

I'm not a robot



I agree to the terms.

Gambar 4. 18 Pembuatan Key API

Penulis melakukan pengisian formulir dan melakukan “Register”

- c. Setelah penulis mendaftar, penulis mendapatkan *API Key* yang dimana *API Key* ini akan digunakan untuk mendapatkan daftar berita yang akan penulis kembangkan pada *chatbot* layanan berita.

Registration complete

Your API key is: 38fa7f8bc2ec44df8a1ac66f21db96ff

For help getting started please look at our [getting started guide](#).

We post API status updates and other news on our Twitter feed, so please follow us there if that's important to you:

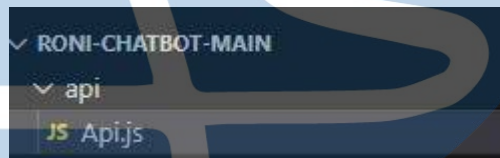
[Follow @NewsAPIorg](#)

My account

Gambar 4. 19 Pembuatan Key API

4. *File Config API*

- a. Pembuatan *file Api.js* yang akan disimpan kedalam folder/*api* di dalam *project vite* yang sudah dibuat sebelumnya oleh penulis



Gambar 4. 20 Persiapan File Config API

STT - NF

b. *File API.js*

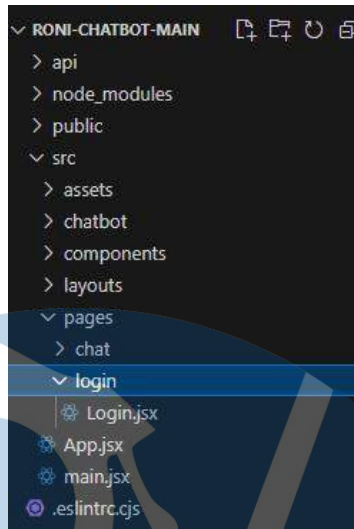
Tabel 4.2 Script File API.js

```
1. import axios from 'axios';
2.
3. let url = `https://newsapi.org/v2/`;
4. let apiKey = `e9d27ee4b8524bce891a5db0896d2e68`;
5. let category = 'everything';
6.
7. class Api {
8.
9.   static async getNews(keyword) {
10.    try {
11.      const response = await
12.      axios.get(`${url}${category}?q=${keyword}&apiKey=${apiKey}&language=id`);
13.      return response.data;
14.    } catch (error) {
15.      console.error(error);
16.    }
17.  }
18.
19.   static async setCategory(category) {
20.     category = category;
21.     return true;
22.   }
23. }
24.
25. export default Api;
```

Perancang menggunakan *dependency* tambahan yaitu “*axios*” yang dimana ini dilakukan untuk mengirim permintaan akses API ke *NewsAPI* beserta *Api Key* yang sudah dibuat sebelumnya.

5. Halaman *Login*

- a. Pembuatan *file* bernama *Login.jsx* yang akan disimpan kedalam folder */src/pages/login* didalam folder project *vite* yang sudah dibuat sebelumnya pada perancangan halaman *login*.



Gambar 4. 21 Persiapan Halaman Login

b. File *Login.jsx*

Tabel 4.3 File *Login.jsx*

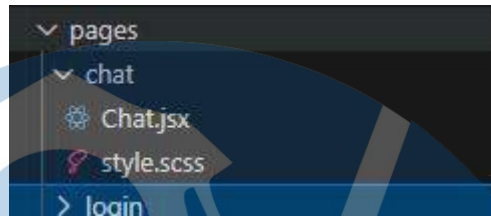
<ol style="list-style-type: none"> 1.2.3.4.5.6.7.8.9.10.11.12.13.14.15.16.17.18.19.20. 	<pre> const Login = () => { const setSession = () => { sessionStorage.setItem('sessionChat', true); checkSession(); } const checkSession = () => { if(sessionStorage.getItem('sessionChat')) { window.location.href = '/chat'; } } return(<> <div className="row g-0 justify-content-center bg- body-dark"> </div> </>) }; export default Login; </pre>
--	--

Script yang ada didalam file *Login.jsx* ini terdiri dari *File HTML Login*, “*checkSession*” dan “*setSession*”. Fungsi dari *checkSession* ini untuk mengecek jika pengguna sudah menjalankan sesi maka pengguna tidak berhak untuk mengakses kembali ke halaman *Login*, melainkan langsung *redirect* ke halaman *Chat*. Lalu fungsi *setSession* untuk

mendaftarkan sesi baru untuk masuk ke halaman *Chat*.

6. Halaman *Dashboard Chat*

- a. Pembuatan *File* bernama *Chat.jsx* yang penulis simpan ke dalam folder */src/pages/chat* untuk melakukan eksekusi pada halaman *dashboard*



Gambar 4. 22 *Persiapan Halaman Dashboard Chat*

- b. Dalam *file dashboard chat*, ada beberapa fungsi/method penting yaitu
 - i. Fungsi pengecekan sesi seperti *script* dibawah ini:

Tabel 4.4 *Fungsi pengecekan sesi*

1.	<code>const News = () => {</code>
2.	
3.	<code>const sessionChat =</code>
4.	<code>sessionStorage.getItem('sessionChat');</code>
5.	<code>if(!sessionChat) {</code>
6.	<code>window.location.href = '/login';</code>
7.	<code>}</code>

Fungsi pengecekan sesi ini bisa dikatakan sebagai *middleware* untuk mencegah *user* mengakses *chat* tanpa melalui halaman *login*. Jika *user* belum memiliki sesi maka akan diarahkan ke halaman *Login*

- ii. *Script* Fungsi penghapusan sesi

Tabel 4.5 *Fungsi pengecekan sesi*

1.	<code>const destroySession = () => {</code>
2.	<code>sessionStorage.removeItem('sessionChat');</code>
3.	<code>window.location.href = '/login';</code>
4.	<code>}</code>
5.	

Fungsi ini akan menjalankan perintah untuk menghapus sesi, fungsi ini akan dikaitkan dengan tombol *Logout*. Jika *user* klik tombol *Logout* makaan fungsi ini akan dijalankan lalu akan diarahkan ke halaman *Login*.

iii. Script Fungsi Current Step

Tabel 4.6 Fungsi Current Step

1.	<code>const [currentStep, setCurrentStep] = useState([</code>
2.	<code> {</code>
3.	<code> name: 'selectingCategoryButton',</code>
4.	<code> id: '1',</code>
5.	<code> active: false,</code>
6.	<code> }, {</code>
7.	<code> name: 'selectingCategoryCustom',</code>
8.	<code> id: '2',</code>
9.	<code> active: false,</code>
10.	<code> }</code>
11.	<code>]);</code>

Fungsi ini merupakan bagian terpenting untuk menentukan langkah dari respon yang akan dikirimkan melalui *chatbot*.

iv. Script Step Filtering

Tabel 4.7 Step Filtering

1.	<code>if(currentStep[1].active !== true) {</code>
2.	<code> setTimeout(() => {</code>
3.	<code> setMessages(previousMessages</code>
4.	<code> [...previousMessages, {</code>
5.	<code> id: nanoid(),</code>
6.	<code> message: 'Maaf keyword yang anda ketik tidak</code>
7.	<code>valid',</code>
8.	<code> user: false,</code>
9.	<code> }));</code>
10.	<code> }, 500);</code>
11.	<code> return false;</code>
12.	<code>}</code>

Fungsi ini akan mengecek inputan dari *user*, jika *user* menginput perintah yang tidak sesuai dengan step pada *index* ke-1 (*selectingCategoryCustom*), maka *bot* akan mengirimkan pesan “Maaf keyword yang ada ketik tidak valid”

v. *Script Fungsi Pencarian Berita berdasarkan Keyword dari User*

Tabel 4.8 Fungsi

```
1. setTimeout(() => {
2.     setMessages(previousMessages => [...previousMessages,
3.     {
4.         id: nanoid(),
5.         message: `Berikut berita serupa dengan keyword
6. <strong>`${message}`</strong> berdasarkan kategori yang
7. dipilih sebelumnya`,
8.         user: false,
9.     }]);
10.
11.     Api.getNews(message).then(response => {
12.         const articles = response.articles;
13.         setMessages(previousMessages => [...previousMessages,
14.         {
15.             id: nanoid(),
16.             message: null,
17.             user: false,
18.             articles: articles,
19.         }]);
20.
21.         setCurrentStep(previousStep =>
22.             previousStep.map(step =>
23.                 step.name === 'selectingCategoryCustom'
24.                 ? {...step, active: false}
25.                 : {...step, active: false}
26.             )
27.         )
28.
29.         setTimeout(() => {
30.             setMessages(previousMessages =>
31.                 [...previousMessages, {
32.                     id: nanoid(),
33.                     message: `Apakah kamu ingin memilih kategori lain?`,
34.                     user: false,
35.                     options: [
36.                         { value: 'change-category', label: 'Ganti
37. Kategori'}
38.                     ]
39.                 }]);
40.             }, 500);
41.
42.         })
43.     }, 500)
44.
45.     e.target.reset();
46. }
47.
```

Pada kode ini, ketika *user* menginput *keyword* berita *bot* akan memberikan respon diiringi dengan proses pengambilan data dari API yang sudah perancang jelaskan pada poin sebelumnya. Setelah semua proses selesai, maka *bot* akan mengirimkan respon kembali berupa pesan “Apakah kamu ingin memilih kategori lain?” diikuti dengan tombol “Ganti Kategori” jika *user* ingin mengganti kategori/mereset ulang *prompt*.

4.3 Implementasi Antarmuka

Pada implementasi antarmuka penelitian merupakan tahapan dimana peneliti menampilkan tampilan hasil dari aplikasi yang dibuat yaitu aplikasi pengembangan *chatbot* pada layanan berita berbasis web menggunakan *framework react*, yang mencakup tampilan *login*, *chat*, dan pilihan berita .

4.3.1 Halaman Login



Gambar 4. 23 Halaman Login

Pada halaman *login* dibagi menjadi dua bagian - sisi kiri berwarna biru tua dan sisi kanan berwarna putih.

Sisi Kiri:

- Logo "RC" terlihat di pojok kiri atas.
- Teks "Bot RoniNews" sebagai judul.
- Pesan sambutan: "Selamat datang di RoniNews! silahkan *login* untuk melanjutkan"

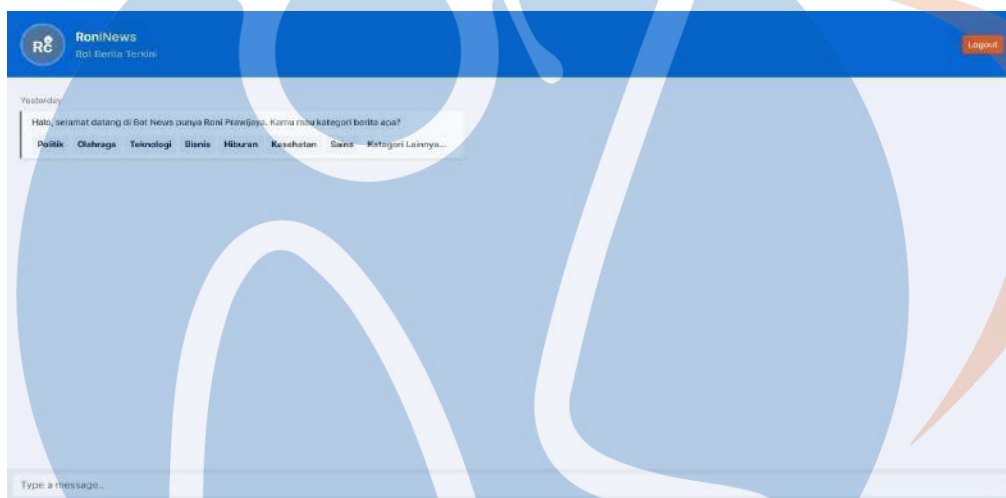
- Gambar latar belakang berupa pemandangan gunung.

Sisi Kanan:

- Nama aplikasi "RoniNews" di bagian atas dengan kata "MASUK" di bawahnya.
- *login* Tombol "Masuk sebagai Tamu".

Pada gambar diatas terdapat halaman *login*, dimana pada halaman *login* ini dirancang untuk memberikan pengalaman sederhana dan mudah bagi pengguna.

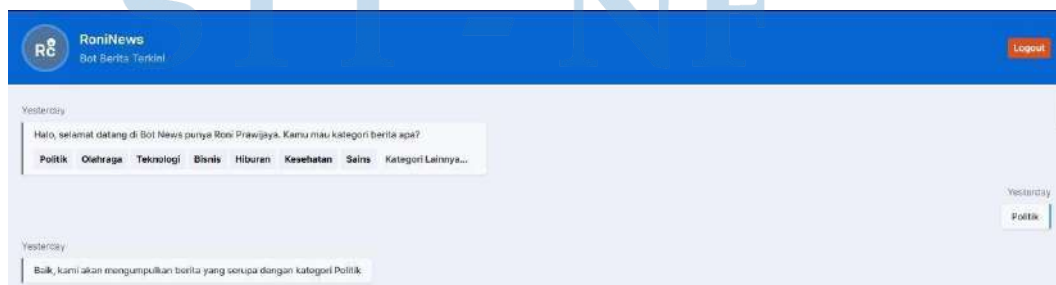
4.3.2 Halaman chat login tamu / guest



Gambar 4. 24 Halaman Chat Login Tamu

Pada gambar di atas memperlihatkan proses *login* ke *chatbot user* bisa melakukan *login* halaman *login* tamu/guest menampilkan halaman *chat* dan bisa melihat kategori berita saat baru pertama masuk ke halaman *dashboard*.

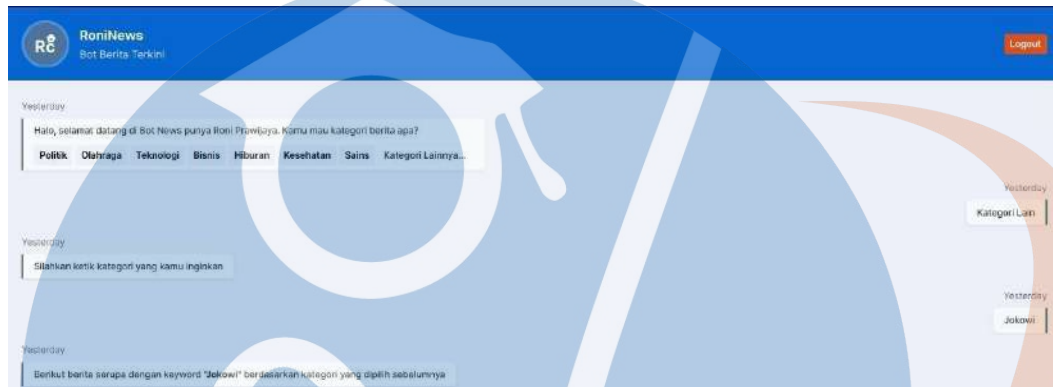
4.3.3 User memilih kategori yang disediakan oleh bot via tombol CTA (Call to Action)



Gambar 4. 25 Halaman User Memilih Kategori

Pada gambar diatas dijelaskan bahwa *User* bisa memilih kategori yang disediakan oleh bot melalui CTA (*Call to Action*) dimana bot akan membalas pesan dengan *keyword* yang dipanggil oleh *user* sehingga bot dapat membalas *keyword* tersebut.

4.3.4 Tampilan ketika user memilih kategori custom dari tombol "Kategori Lainnya"



Gambar 4. 26 Halaman User Memilih Kategori Custom

Tampilan *user* ketika memilih kategori *costume* dari tombol "Kategori Lainnya" *user* bisa mengetik pada laman papan ketik *bot* untuk memilih berita *custom* dari tombol *message*, sehingga *user* dapat mengetik apa saja yang diinginkan dan *bot* akan membalas jika *keyword* tersebut sesuai.

4.3.5 Tampilan jika berita tidak sesuai/tidak ada

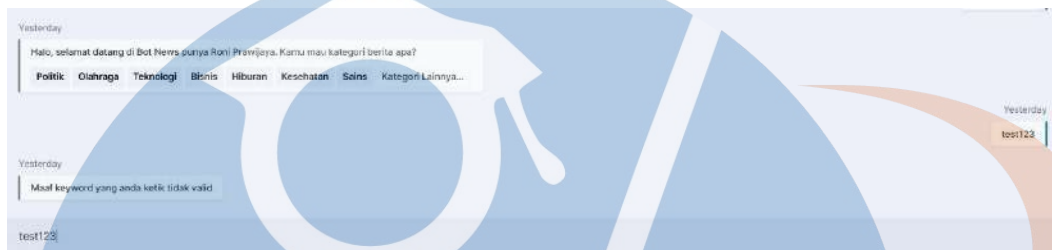


Gambar 4. 27 Halaman Tampilan Jika Berita tidak Sesuai/Tidak Ada

Pada gambar diatas adalah tampilan jika *user* memilih atau mengetik

keyword yang tidak sesuai dan tidak dimengerti oleh *bot*, maka *user* akan mendapatkan *message error* “*Maaf, kami tidak menemukan berita yang sesuai dengan kategori yang kamu pilih. Silahkan coba kategori lain atau ketik keyword yang kamu inginkan*”. Dari pesan tersebut *bot* akan mengarahkan *user* untuk memilih kategori lain dan mengetik *keyword* yang diinginkan kembali.

4.3.6 Tampilan ketika keyword/prompt yang diketik user tidak sesuai



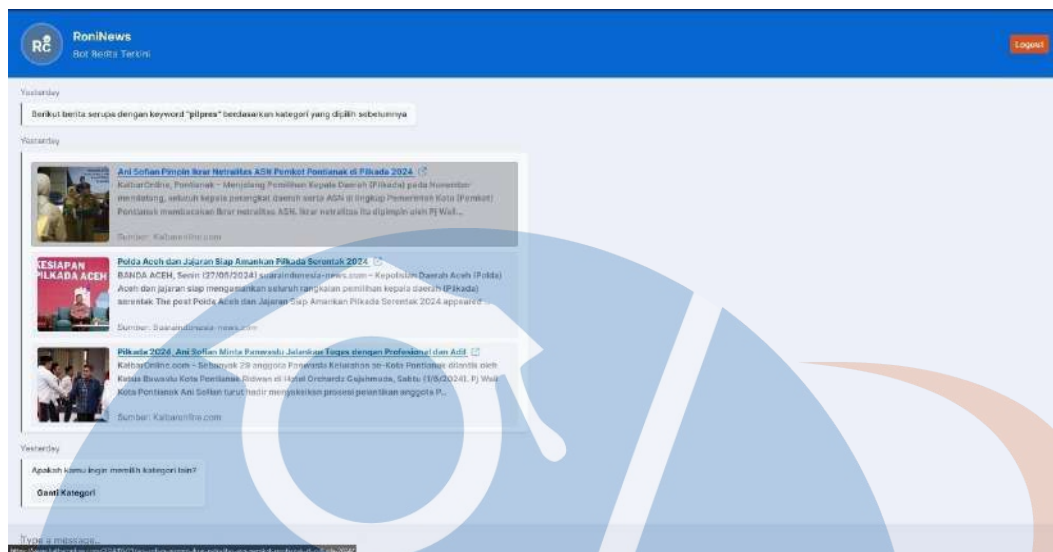
Gambar 4. 28 Halaman Tampilan Ketika Keyword Tidak Sesuai

Pada gambar diatas menunjukkan percakapan *bot* berita dengan penjelasan sebagai berikut :

- Bot menyapa pengguna dengan "Halo, selamat datang di Bot News punya Roni Prawijaya." dan menanyakan kategori berita yang diinginkan.
- Bot menampilkan beberapa pilihan kategori berita: Politik, Olahraga, Teknologi, Bisnis, Hiburan, Kesehatan, Sains, dan opsi "Kategori Lainnya...".
- Pengguna merespons dengan mengetik "test123".
- Bot merespons "*Maaf keyword yang anda ketik tidak valid*", menunjukkan bahwa input "*test123*" tidak dikenali sebagai pilihan kategori yang valid.
- Pengguna mengetik "*test123*" lagi.

Percakapan ini menggambarkan interaksi awal user dengan bot berita yang meminta user memilih kategori, namun user memberikan input yang tidak sesuai dengan pilihan yang tersedia sehingga *respon* yang diberikan bot adalah “*Maaf keyword yang anda ketik tidak valid*” .

4.3.7 Tampilan Hasil Pencarian Berita



Gambar 4. 29 Halaman Tampilan Hasil Pencarian Berita

Pada gambar diatas adalah gambar hasil akhir dari pencarian berita *bot* akan membalas dan memberikan berita yang diinginkan dengan contoh pada *capture* di atas, sehingga *user* dapat membaca dan memilih berita mana yang diinginkan dan berita mana yang diinginkan oleh *user* sesuai *keyword* yang diketik.

4.3.8 Tampilan Logout



Gambar 4. 30 Halaman Tampilan Logout

Pada gambar diatas terdapat menu *logout* yang muncul pada bagian kanan halaman, pengguna dapat *logout* saat pencarian sudah selesai dan sudah membaca berita, maka tampilan akan kembali pada menu *login* dan tampilan

awal *login* seperti gambar di atas.

4.4 Evaluasi dan Pengujian

Evaluasi dan pengujian merupakan tahapan penting dalam merancang dan mengembangkan sistem, tujuan dari proses evaluasi dan pengujian itu sendiri adalah untuk memastikan bahwa *chatbot* yang yang dirancang sesuai dengan kebutuhan pengguna dan sistem berfungsi sesuai yang diharapkan dengan benar.

4.4.1 Evaluasi Sistem

Evaluasi sistem adalah proses mengevaluasi dan menguji sistem yang dirancang untuk menentukan apakah aplikasi bekerja sesuai dengan kebutuhan pengguna. Pengujian dilakukan dengan menggunakan *black box testing* namun nantinya *user* akan mendapatkan beberapa tantangan masalah diantaranya yaitu:

1. Masalah koneksi

Masalah konektivitas umumnya terjadi ketika ada gangguan atau kesalahan saat mengakses atau menggunakan jaringan komunikasi, seperti Internet atau jaringan area lokal (LAN). Hal ini dapat disebabkan oleh berbagai faktor, termasuk perangkat keras yang rusak, perangkat lunak yang tidak berfungsi, gangguan infrastruktur jaringan, dan masalah konfigurasi. Oleh karena itu, permasalahan jaringan menjadi tantangan dalam mengoperasikan sistem yang kami rancang dan Web.

2. Berita yang tidak *real time update*

Berita yang disajikan bergantung dari pihak ketiga (*NewsAPI*) sehingga aplikasi yang penulis rancang memiliki keterbatasan pada berita terbaru dan berita yang diinginkan oleh *user*.

3. Limitasi Akses API (pihak ketiga)

Limitasi akses API dari pihak ketiga mengacu pada pembatasan atau batasan yang diterapkan oleh penyedia layanan API (*Application Programming Interface*) terhadap penggunaan dan akses terhadap API pihak ketiga yang memiliki batasan *request* sebanyak 200 hit/jam.

4.4.2 Pengujian Sistem Black Box Testing

Tabel 4 9 Black Box Testing

No	Pengujian	Ekspektasi	Hasil
1.	User bisa melihat halaman <i>login</i>	User bisa melakukan <i>login</i> menggunakan yaitu <i>login</i> sebagai tamu	Berhasil
2.	User Bisa melihat halaman chat	User bisa melakukan interaksi dengan Bot	Berhasil
3.	User bisa memilih kategori berita	User memilih kategori yang sudah disediakan ataupun memilih kategori sesuai keinginan	Berhasil
4.	User bisa melihat hasil berita	Berita yang sesuai kategori/ <i>keyword</i> yang dipilih oleh <i>user</i> akan ditampilkan	Berhasil
5.	User dapat mengganti kategori/ <i>keyword</i>	User dapat memilih tombol ganti kategori	Berhasil
6.	Bot menampilkan pesan <i>error</i>	Bot menampilkan pesan error Ketika <i>keyword</i> / <i>kategori</i> yang dipilih tidak tersedia	Berhasil
7.	<i>Logout</i>	User dapat mengakhiri sesi dengan klik tombol <i>logout</i>	Berhasil

Pada pengujian *Black Box* memiliki 7 skenario pengujian yang dilakukan perancang aplikasi. Pada jumlah pengujian yang sukses didapatkan sebanyak 7 keberhasilan, sedangkan yang tidak berhasil yaitu 0, sehingga persentase keberhasilan dapat disimpulkan sebagai berikut :

1. Berhasil : $(7/7) \times 100\% = 100\%$
2. Tidak berhasil : $(0/0) \times 100\% = 0\%$

BAB V

KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari seluruh proses yang dilakukan dalam penelitian ini dan saran-saran yang menjadi masukan untuk pengembangan selanjutnya.

5.1 Kesimpulan

Penelitian ini bertujuan untuk melakukan perancangan *chatbot* pada layanan berita berbasis web dengan menggunakan *framework react*, pada penelitian ini bisa didapatkan sebuah kesimpulan sebagai berikut :

1. Pada pengembangan *chatbot* penyedia layanan informasi berita berbasis web dengan menggunakan *framework React* merupakan langkah yang strategis dan relevan pada era digital saat ini. Berdasarkan analisis terhadap rancangan *chatbot* yang penulis kembangkan, *React* terbukti menawarkan fleksibilitas dan efisien dalam merancang antarmuka yang responsif serta interaktif. Dengan kemampuan *React* dalam mengelola komponen-komponen *UI* secara efisien, rancangan *Chatbot* dapat dioptimalkan untuk memberikan pengalaman pengguna yang lebih baik dan memudahkan integrasi dengan layanan berita secara *real-time*.
2. Dari pengujian yang dilakukan dapat disimpulkan bahwa pengembangan *chatbot* penyedia berita berbasis web menggunakan *framework react* berjalan dengan efektif.

5.2 Saran

Pada perancangan *chatbot* penyedia informasi berita berbasis web menggunakan *framework react*, perancang menyadari masih banyak kekurangan sehingga masih memerlukan pengembangan lebih lanjut. Terdapat beberapa poin mengenai rancangan penelitian yang akan datang seperti berikut:

1. *chatbot* pada informasi layanan berita berbasis web menggunakan *framework react* belum mendukung tanggapan berupa gambar atau dokumen, sehingga penulis berharap untuk pengembang dapat menambahkan fitur tanggapan *chatbot* berupa gambar dan dokumen serta tidak harus memilih kategori namun *user* bisa langsung interaksi dengan *chatbot*.
2. Pada *Chatbot* informasi layanan berita menggunakan *framework react* yang penulis rancang belum *support* untuk *user* bisa *login* menggunakan *email*, dikarenakan belum ada *database* terkait *chatbot* yang penulis rancang, namun *user* bisa *login* sebagai tamu/*guest*, Sehingga penulis berharap kepada pengembang untuk dapat mengembangkan *chatbot* layanan berita ini dengan menyediakan *database* dan *login* menggunakan *email*.
3. Penulis berharap kepada pengembang terkait *chatbot* pada informasi layanan berita berbasis web menggunakan *framework react* agar *chatbot* lebih memahami, merespons pesan, dan memberikan jawaban yang lebih akurat dan relevan.

DAFTAR REFERENSI

- [1] R. N. Astuti and D. M. Fatchan, “Perancangan Aplikasi Teknologi Chatbot Untuk Industri Komersial 4.0,” Bekasi: Prosiding Seminar Nasional Teknologi dan Sains (SNasTekS), Sep. 2019.
- [2] E. Adamopoulou and L. Moussiades, “Chatbots: History, technology, and applications,” *Machine Learning with Applications*, vol. 2, p. 100006, Dec. 2020, doi: 10.1016/j.mlwa.2020.100006.
- [3] G. Caldarini, S. Jaf, and K. McGarry, “A Literature Survey of Recent Advances in Chatbots,” *Information (Switzerland)*, vol. 13, no. 1, Jan. 2022, doi: 10.3390/info13010041.
- [4] E. Adamopoulou and L. Moussiades, “An Overview of Chatbot Technology,” in *IFIP Advances in Information and Communication Technology*, Springer, 2020, pp. 373–383. doi: 10.1007/978-3-030-49186-4_31.
- [5] P. D. Larasati, A. Irawan, S. Anwar, M. F. Mulya, M. A. Dewi, and I. Nur Fatima, “Chatbot helpdesk design for digital customer service,” *Applied Engineering and Technology*, vol. 1, no. 3, pp. 138–145, Apr. 2022, doi: 10.31763/aet.v1i3.684.
- [6] R. Kurniawan, M. Zulfikar, P. Studi Teknik Informatika, and S. Tinggi Manajemen Informatika dan Komputer Indodaya Suvana, “Perancangan Aplikasi Chatbot Pada Salah Satu Industri E-Commerce Di Jakarta.”
- [7] Z. Zhang, X. Zhang, and L. Chen, “Informing the Design of a News Chatbot,” in *Proceedings of the 21st ACM International Conference on Intelligent Virtual Agents, IVA 2021*, Association for Computing Machinery, Inc, Sep. 2021, pp. 224–231. doi: 10.1145/3472306.3478358.
- [8] O. E. Nordberg and F. Guribye, “Conversations with the News: Co-speculation into Conversational Interactions with News Content,” in *Proceedings of the 5th International Conference on Conversational User Interfaces, CUI 2023*, Association for Computing Machinery, Inc, Jul. 2023. doi: 10.1145/3571884.3597123.
- [9] E. V. i Pereña, “NetIQ Idm Front-End Implementation With React.Js & Development Of A User Support Chatbot With Python,” A Degree Thesis, Universitat Politècnica de Catalunya, Barcelona, 2023.

- [10] R. Koçi, X. Franch, P. Jovanovic, and A. Abelló, “Web API evolution patterns: A usage-driven approach,” *Journal of Systems and Software*, vol. 198, Apr. 2023, doi: 10.1016/j.jss.2023.111609.
- [11] C. González-Mora, C. Barros, I. Garrigós, J. Zubcoff, E. Lloret, and J. N. Mazón, “Improving open data web API documentation through interactivity and natural language generation,” *Comput Stand Interfaces*, vol. 83, Jan. 2023, doi: 10.1016/j.csi.2022.103657.
- [12] A. K. Chhetri, “Developing a Front-end web app using React,” Bachelor’s thesis, Haaga-Helia University of Applied Sciences, 2024.
- [13] S. L. Kekurangan *et al.*, “Literature Study Of The Lack And Excess Of Testing The Black Box,” *Teknomatika*, vol. 10, no. 02, pp. 1–5, 2020.
- [14] D. Felicio, J. Simao, and N. Datia, “Rapitest: Continuous black-box testing of restful web apis,” in *Procedia Computer Science*, Elsevier B.V., 2023, pp. 537–545. doi: 10.1016/j.procs.2023.01.322.
- [15] R. N. Astuti and D. M. Fatchan, *Prosiding Seminar Nasional Teknologi dan Sains (SNasTekS)*. 2019.
- [16] B. Huberta and A. B. Wijaya, “Perancangan Chatbot Website Program Studi Informatika Menggunakan Framework Codeigniter,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 11, no. 3, Aug. 2023, doi: 10.23960/jitet.v11i3.3225.
- [17] A. Ardiansyah, “Pendampingan Perancangan Chatbot Sebagai Media Interaktif Dalam Menghadapi Tantangan Era Digitalisasi,” *Lamahu: Jurnal Pengabdian Masyarakat Terintegrasi*, vol. 2, no. 1, pp. 44–55, Feb. 2023, doi: 10.34312/ljpmt.v2i1.18078.
- [18] M. R. Herfian and A. R. Adriansyah, “Analisis Dan Perancangan Aplikasi Chatbot Dalam Pelayanan Penerimaan Mahasiswa Baru Pada Perguruan Tinggi,” *Jurnal Informatika Terpadu*, vol. 2, pp. 87–93, 2021.