

BAB II

KERANGKA KONSEP

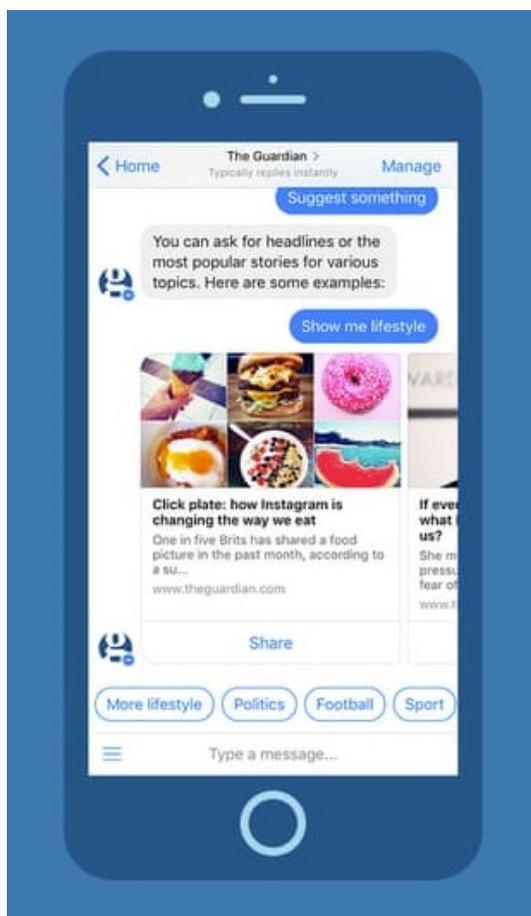
2.1 Tinjauan Karya Sejenis

Sebagai bahan acuan, penulis mencari informasi dari karya yang sudah pernah dibuat sebelumnya mengenai *chatbot*. Penulis mengelompokkan karya sejenis menjadi dua kategori yaitu media yang menggunakan *chatbot* dan referensi konsep atau jurnal mengenai pengembangan sebuah *chatbot* dalam ranah jurnalistik. Beberapa karya sejenis yang ditinjau penulis dari segi media antara lain: *The Guardian*, *CNN*, dan *Quartz*. Rujukan pertama adalah jurnal yang ditulis oleh Andreas Veglis dan Theodora A. Manio pada 2019 yang berjudul “Chatbots on the Rise: A New Narrative in Journalism.” Jurnal ini menjelaskan bagaimana perkembangan teknik otomasi dilihat sebagai inovasi baru dalam jurnalistik terutama dalam platform media berita. Gelombang otomasi baru menggabungkan kecerdasan buatan dengan teknologi kognitif yang membantu dalam bidang jurnalisme yang awalnya dinilai dapat mengurangi bahkan menghilangkan peran jurnalis dalam melakukan tugasnya yaitu menyebarkan berita di era data besar.

Penulisan ini menjelaskan kelebihan dan kekurangan penggunaan *chatbot* serta serangkaian masalah bagi lingkungan profesional jurnalistik yang timbul dalam dunia barat seperti masalah kreativitas, etika, dan orisinalitas berita yang berpusat pada argumen bahwa robot tidak akan pernah bisa menggantikan jurnalisme manusia tanpa menimbulkan ancaman nyata terhadap nilai-nilai demokrasi yang diakui secara kritis. Namun, disisi lain dijelaskan bahwa *chatbot* dapat membebaskan jurnalis dari pekerjaan rutin mereka dalam kuantitas produksi berita dan memungkinkan mereka untuk berkonsentrasi pada kualitas, membebaskan kapasitas untuk analisis dan pelaporan yang mendalam. Jurnalisme *chatbot*, sebagai transformasi logis robot dan jurnalisme komputasi melahirkan peluang sekaligus tantangan baru, baik bagi khalayak maupun bagi profesi jurnalistik. Narasi jurnalistik baru ini dapat menawarkan solusi yang realistis dan layak untuk masalah yang dihadapi jurnalis di era *big data*, asalkan para profesional dan entitas media berita mengelola proses tersebut di ruang redaksi.

Tinjauan karya sejenis lainnya juga dibahas dalam penulisan pertama, yaitu media *The Guardian* yang sudah tertarik pada bot sejak tahun 2010 yang dapat membuat berita olahraga secara otomatis dan memungkinkan mereka memperoleh statistik permainan dan informasi historis tim serta pemain. Kemudian, bot menggabungkan data statistik permainan dan frasa menjadi sebuah berita. Empat tahun setelahnya, Guarbot hadir untuk melengkapi informasi keuangan dengan data yang

kompleks. Dalam artikel *The Guardian* yang berjudul “Introducing the Guardian Chatbot” menjelaskan mengenai inisiatif *chatbot* yang dapat berkomunikasi dengan pengguna melalui Facebook Messenger dengan layanan yang tersedia seperti mengirimkan pesan kunci tertentu. Beberapa layanan yang disediakan adalah langganan berita pagi, memberikan lima berita teratas tentang topik-topik tertentu, dan lain-lain.



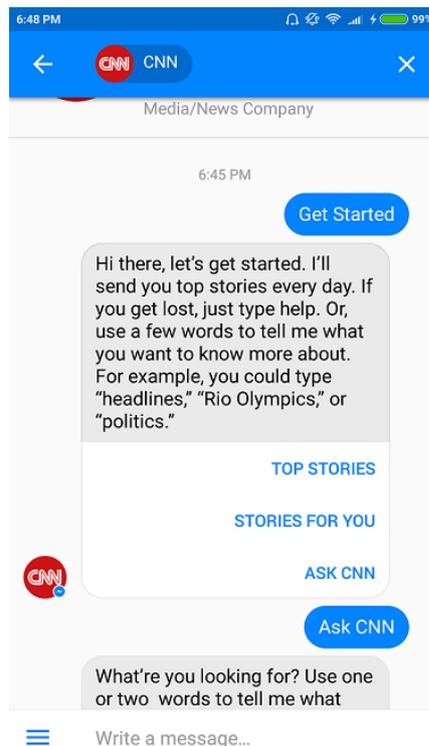
Sumber: Theguardian.com, 2016.

Gambar 2.1 Tampilan Bot The Guardian

Karya sejenis lainnya adalah jurnal yang ditulis oleh Bronwyn Jones and Rhianne Jones dengan judul “Public Service Chatbots: Automating Conversation with BBC News.” Studi ini menjelaskan bahwa *BBC* telah bereksperimen dengan bot, khususnya *chatbots* untuk distribusi berita mengenai layanan publik daripada membangun interaksi dengan audiens. Bot *BBC* dinilai belum stabil dan belum memberikan gambaran bagaimana bot sedang dimobilisasi untuk menguji bentuk distribusi berita baru yang dipersonalisasi dengan berinteraksi secara dua arah yang mencerminkan nada percakapan populer yang lazim media sosial sehingga penulis menyarankan adanya narasi organisasi bersama antara teknologi dan jurnalis untuk meningkatkan jangkauan keterlibatan, terutama masyarakat yang kurang dilayani termasuk anak muda. Bot *BBC*, pada saat penulisan, tidak digerakan oleh *Artificial Intelligence* (AI) dan *Machine Learning* (ML). Namun, studi ini menangkap momen penting dalam perubahan wajah *BBC News* yang berencana membangun pengalamannya mengembangkan bot ini hingga bereksperimen dengan AI untuk jurnalisme percakapan. Bot, khususnya *chatbots* di *BBC*, dipandang sebagai kesempatan untuk menjangkau dan melibatkan khalayak yang tidak terlayani dengan membuat berita yang lebih mudah diakses oleh kaum muda dan siapa saja.

Mengikuti jejak *The Wall Street Journal*, karya selanjutnya adalah bot dari media *CNN* yang dijelaskan dalam *Marutitech.com* dan *Journalism.co.uk* yang berjudul “‘We curate, you query’: Inside CNN's new Facebook Messenger chat bot.” *CNN* adalah salah satu organisasi berita

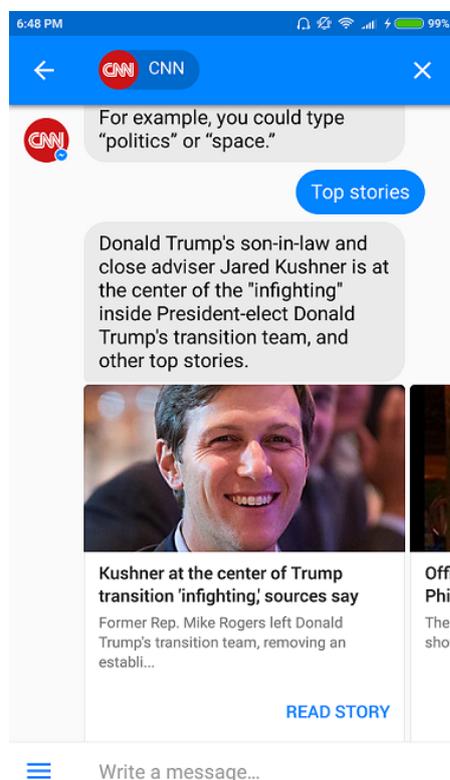
pertama yang mengadopsi platform bot baru di Facebook Messenger. Hal ini merupakan kesempatan untuk menjangkau khalayak global dan terhubung dengan cara yang sangat pribadi dalam skala besar. *CNN* menggunakan *chatterbots* untuk berinteraksi dengan pengguna secara alami seperti manusia. Pelanggan *CNN* dapat menerima pemberitahuan harian tentang berita utama. *Chatbot* berita akan merekomendasikan konten yang dipersonalisasi berdasarkan preferensi pengguna dari minat dan aktivitas sebelumnya. *Chatbot* ini juga memberikan pengalaman yang lebih personal kepada setiap pengguna. Tentunya, hal ini dinilai dapat meningkatkan keterlibatan dan retensi pelanggan.



Sumber: Marutitech.com

Gambar 2.2 Bot CNN Messenger (Memberikan Opsi Seperti *Top Stories*, *Stories for you*, dan *Ask CNN*)

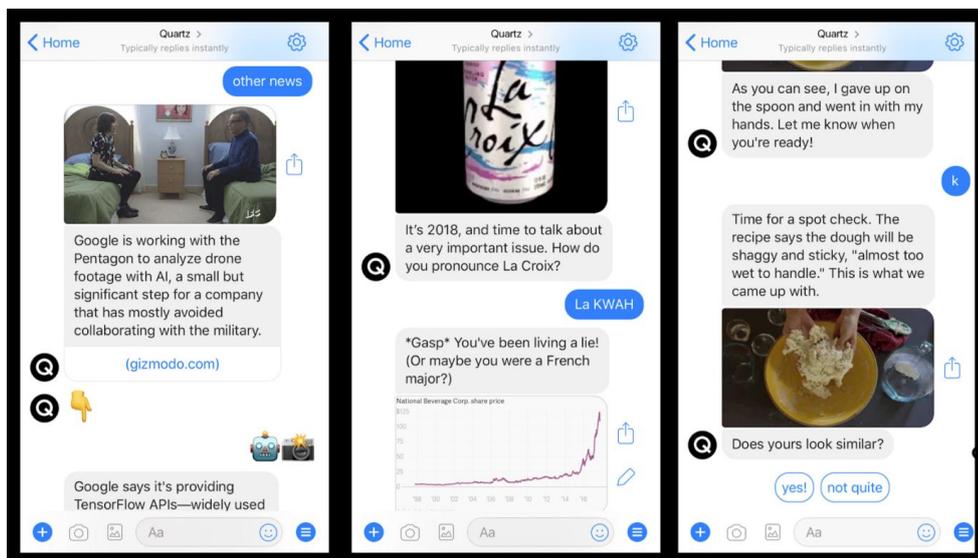
Secara spesifik *CNN* memberikan tiga opsi ketika pengguna mulai menggunakan bot tersebut yang kemudian akan merespon pengguna per pilihan untuk mendapatkan preferensi informasi yang membuat pola personalisasi. Salah dua dari ketiga opsi tersebut adalah *Stories for you* dan *Ask CNN*. *Stories for you* adalah berita yang dipersonalisasi untuk setiap pengguna dengan memberikan pilihan artikel berdasarkan konten populer *CNN* dan apa yang telah dikonsumsi individu tersebut sebelumnya. Kisah-kisah tersebut dapat dibaca secara keseluruhan di jendela web seluler terpisah atau diringkas oleh bot dalam beberapa paragraf di Messenger. *Ask CNN* mengharuskan pengguna untuk mengetikkan kata kunci tertentu untuk menerima informasi tentang topik tertentu.



Sumber: Marutitech.com

Gambar 2.3 Memilih Berita Utama, Bot CNN Merespons dengan Berita Teratas

Hampir sama dengan tinjauan karya sebelumnya, artikel dalam *Journalism.co.uk* yang berjudul “Personality is everything: How Quartz made its chatbot more 'human',” *Quartz* memilih untuk menyampaikan berita melalui percakapan antarmuka (alias bot) yang memungkinkan pengguna menanggapi pesan dan pembaruan dengan mengetuk tombol dan emoji yang muncul di bagian bawah layar. Zach Seward selaku Wakil Presiden Senior Bidang Produk dan Editor Eksekutif *Quartz* mengatakan bahwa meskipun aplikasi ini bertujuan untuk membuat orang-orang mengetahui berita ekonomi global terbaru, aplikasi ini juga merupakan pengalaman hiburan. Aplikasi ini juga akan menampilkan pertanyaan kuis. Dalam artikel *Quartz* yang ditulis olehnya sendiri (2018), Quartz menyuguhkan empat menu, yaitu *Obsessions*, *Projects*, *News*, dan *Messages*.



Sumber: Qz.com, 2018.

Gambar 2.4 Tampilan Bot Quartz

2.2 Teori dan Konsep

2.2.1 Discord

Discord adalah aplikasi obrolan suara, video, dan teks gratis yang digunakan oleh puluhan juta orang berusia 13 ke atas untuk berbicara dan bergaul dengan komunitas atau teman (Discord.com, n.d., para. 1). Discord yang diluncurkan pada 2015 secara orisinal dibentuk untuk para *gamers* yang kemudian berkembang untuk seluruh komunitas di internet dengan tujuan membangun komunitas dan komunikasi seperti membagikan pesan, gambar, video, dan audio (Delfino & Dean, 2021, paras. 3-4). Lebih lanjut dijelaskan, Discord memiliki fitur-fitur seperti *server* yang berisi saluran teks (*text channel*) dan saluran suara (*voice channel*) yang bisa dikostumisasi sesuai dengan topik atau pembicaraan tertentu. Seperti aplikasi lainnya, Discord juga menyediakan fitur pesan langsung, panggilan suara, dan panggilan video dengan sesama pengguna. Untuk meningkatkan pengalaman pengguna, Discord dapat dihubungkan dengan aplikasi lain seperti Youtube, Spotify, Steam, dan lain-lain.

Berikut beberapa istilah dan penjelasan lebih lanjut mengenai fitur-fitur yang disediakan dalam aplikasi Discord (Discord.com, n.d., paras. 5-13).

1. *Server*: *Server* adalah ruang di Discord yang dapat dibuat untuk komunitas atau kelompok tertentu. *Server* dapat bersifat publik maupun privat.
2. *Channel*: *Channel* atau saluran aplikasi Discord dibagi menjadi *channel* teks dan suara yang biasanya didedikasikan untuk topik tertentu dan dapat memiliki aturan yang berbeda. Di saluran teks, pengguna dapat mengirimkan pesan, mengunggah file, membagikan gambar, gif, dan video. Di saluran suara, pengguna dapat terhubung melalui panggilan suara atau video secara *real time*, serta dapat melakukan *sharing screen* atau yang dikenal dengan *Go Live* bersama teman.
3. *Direct Messages (DMs)* dan *Group Direct Messages (GDMs)*: Pengguna dapat mengirim pesan pribadi ke pengguna lain sebagai pesan langsung (*DM*) dan memulai panggilan suara atau video. Pengguna juga memiliki opsi untuk mengundang maksimal sepuluh orang lainnya dengan membuat *DM* grup pribadi.
4. *Siarkan Langsung (Live)*: Pengguna dapat berbagi layar (*sharing screen*) dengan orang lain yang berada di *server* atau *DM*.

5. Nitro: Nitro adalah layanan langganan premium Discord yang menawarkan fasilitas khusus, seperti menggunakan emoji khusus di setiap *server*, batas unggahan file yang lebih tinggi, dan lain-lain.
6. *Server Boosts*: *Server Boosts* memberi *server* fasilitas khusus seperti lebih banyak slot emoji khusus, kualitas video dan suara yang lebih baik, dan kemampuan untuk mengatur kustomasi tautan undangan khusus. *Server Boosts* dapat dibeli dengan Nitro atau dibeli secara terpisah.

2.2.2 *Chatbot*

Menurut Srinivasan et al., *chatbot* atau *conversational bot* adalah sebuah program perangkat lunak (beberapa kasus berupa perangkat keras) yang diprogram secara otomatis untuk dapat berkomunikasi dan berinteraksi dengan bot menggunakan suara atau teks untuk mengakses informasi (2018, p. 3). Selaras dengan definisi Srinivasan et al., Pratt (2017) menjelaskan *chatbot* sebagai layanan yang menyediakan interaksi *chatting interface* dengan perintah suara atau teks dan direpson secara konversasional layaknya berbicara dengan manusia.

Joseph Weizenbaum telah mengembangkan *chatbot* generasi pertama pada 1966 bernama ELIZA, sebuah program komputer yang berperan sebagai psikoterapis dan berhasil menunjukkan kemampuan

komunikasi antara manusia dan komputer dengan *natural language* (Veglis & Maniou, 2019, p. 1). Kemampuannya untuk berkomunikasi terbatas karena hanya membahas topik khusus dan tidak dapat membuat percakapan panjang (Adamopoulou & Moussiades, 2020, p. 2). Pada 1972, PARRY hadir sebagai kemajuan dari ELIZA. PARRY berperan sebagai pasien skizofrenia yang diuji coba pada 1979 oleh lima psikiater melalui *teletype* untuk memutuskan apakah dia adalah seorang program komputer atau pasien skizofrenia sungguhan (Adamopoulou & Moussiades, 2020, p. 2). ALICE (*Artificial Linguistic Internet Computer Entity*) merupakan *chatbot* daring pertama yang terinspirasi oleh ELIZA dengan menggunakan *pattern-matching*. Namun, ALICE tidak memiliki kecerdasan untuk dapat menghasilkan jawaban seperti manusia yang mengekspresikan emosi atau sikap (Adamopoulou & Moussiades, 2020, p. 2).

Awal tahun 2016, evolusi *Artificial Intelligence Technology* mengubah secara dramatis cara orang berkomunikasi dengan mengembangkan *chatbot* di berbagai platform media untuk berbagai layanan seperti pemasaran, perpesanan, perawatan kesehatan, hiburan, pendidikan, dan lain-lain. (Adamopoulou & Moussiades, 2020, p. 3).

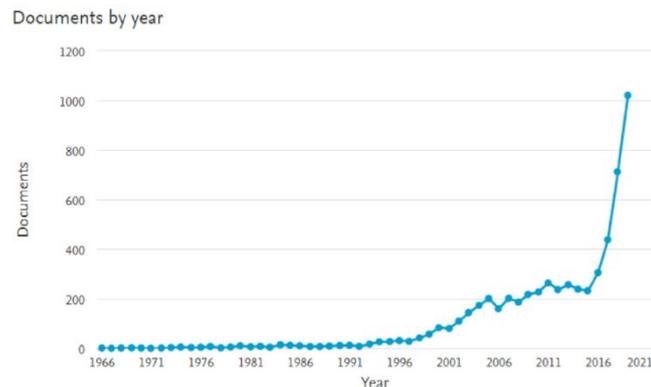


Fig. 1. Search Results in Scopus (*Scopus preview—Scopus—Welcome to Scopus*, 2020), from 1966 to 2019 for the keywords “chatbot” or “conversation agent” or “conversational interface”.

Sumber: Adamopoulou dan Moussiades, 2020.

Gambar 2.5 Peningkatan Penggunaan *Chatbot*, Khususnya Setelah 2016

Adamopoulou dan Moussiades (2020, pp. 3-4) merangkum kategori *chatbot* secara sederhana dengan penjelasan lebih rinci sebagai berikut.

1. *Generic: Chatbot* yang dapat menjawab pertanyaan pengguna apapun dari domain mana saja.
2. *Open Domain: Chatbot* yang dapat beroperasi di lebih dari satu domain.
3. *Closed Domain: Chatbot* yang hanya dapat menjawab pertanyaan tentang domain pengetahuan tertentu.
4. *Interpersonal: Chatbot* yang menawarkan layanan seperti layanan pemesanan di restoran, maskapai penerbangan, dan lain-lain.
5. *Intrapersonal: Chatbot* yang tinggal di domain pengguna dan memahami kebutuhannya. *Chatbot* ini biasanya terhubung ke aplikasi *messenger* seperti Slack dan WhatsApp.

6. *Inter-agent*: *Chatbot* yang menyediakan komunikasi dengan *chatbot* lain.
7. *Chat based/Conversational*: Pengguna berkomunikasi dengan *chatbot* untuk mendapatkan informasi spesifik yang disimpan dalam sumber tetap. *Chatbot* berbasis obrolan atau percakapan memberikan percakapan *natural* dengan pengguna seperti yang dilakukan dengan orang sungguhan.
8. *Task based*: *Chatbot* berbasis tugas menangani berbagai fungsi dan sangat baik dalam meminta informasi dan menanggapi permintaan pengguna seperti pemesanan kamar.
9. Berdasarkan metode *response generation*: *chatbot* berbasis peraturan (*rule-based chatbot*), *retrieval based chatbot*, dan *chatbot* berbasis generatif (*generative based chatbot*).
10. Berdasarkan izin yang diberikan oleh platform pengembangan: *open-source chatbot* dan *chatbot* komersial.
11. Berdasarkan saluran komunikasi: *chatbot* berupa teks, suara, gambar, atau seluruhnya.

Chatbot Categories	Knowledge domain	
		Generic
		Open Domain
		Closed Domain
	Service provided	Interpersonal
		Intrapersonal
		Inter-agent
	Goals	Informative
		Chat based/Conversational
		Task based
	Response Generation Method	Rule based
		Retrieval based
		Generative
	Human-aid	Human-mediated
		Autonomous
	Permissions	Open-source
		Commercial
	Communication channel	Text
		Voice
		Image

Fig. 3. Chatbots' Classification.

Sumber: Adamopoulou dan Moussiades, 2020.

Gambar 2.6 Kategori *Chatbot*

Lebih lanjut Adamopoulou dan Moussiades (2020, pp. 4-7) menjabarkan ada dua pendekatan dalam mengembangkan *chatbot* bergantung pada algoritma dan teknik yang diadopsi.

1. *Pattern matching approaches*

Chatbot berbasis aturan (*rule-based chatbot*) mencocokkan masukan pengguna (*user input*) dengan pola aturan dan jawaban yang telah ditentukan sebelumnya dari sekumpulan tanggapan menggunakan algoritma *pattern matching*. Sistem *chatbot* berbasis aturan biasanya tidak membuat jawaban baru sebagai pengetahuan dalam bentuk percakapan. Setiap respon digunakan sebagai umpan balik untuk memilih jawaban yang sesuai dengan konteks. Semakin luas *database* dengan aturannya, *chatbot* semakin mampu menjawab pertanyaan

pengguna. Kekurangan dari pendekatan *pattern-matching* ini adalah jawaban yang disediakan otomatis, berulang, dan tidak memiliki keaslian spontanitas respon manusia. Beberapa bahasa yang biasanya digunakan untuk implementasi *pattern-matching* approach adalah AIML, Rivescript, and Chatscript. Cleverbot, Chatfuel, dan Watson.

2. *Machine learning approaches*

Chatbot yang mengadopsi *machine learning approaches* mengambil masukan pengguna (*input user*) menggunakan *Natural Language Processing (NLP)* dan tidak memiliki kemampuan untuk mempelajari percakapan. *Chatbot* mempertimbangkan konteks dialog secara keseluruhan. Biasanya, *Artificial Neural Networks (ANNs)* digunakan untuk implementasi *chatbot* ini.

Desain arsitektur *chatbot* yang sesuai berguna untuk mempelajari dan menciptakan *chatbot*. Adamopoulou dan Moussiades (2020, p. 8-10) menyusun ringkasan desain arsitektur dari rangkuman beberapa studi yang bersifat umum dan mencakup detail penting yang berisi komponen fokus utama. Pengembang dapat memutuskan bagian mana yang akan diimplementasikan tergantung pada *chatbot* yang akan dibuat.

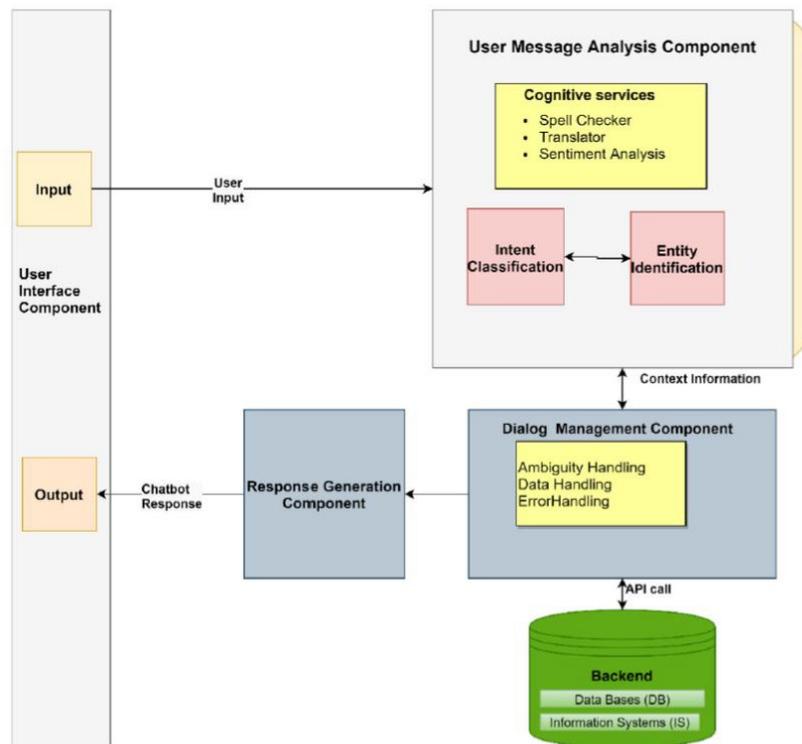


Fig. 8. General Chatbot architecture.

Sumber: Adamopoulou dan Moussiades, 2020.

Gambar 2.7 Arsitekur Umum Pembuatan *Chatbot*

Adamopoulou dan Moussiades (2020, pp. 8-10) merangkum beberapa studi dan menyederhanakan desain arsitektur chatbot tanpa menghilangkan detail komponen utama dan bagian yang berisi fokus penting. Developer apat memutuskan bagian mana yang dapat diimplementasikan, tergantung pada jenis *chatbot*.

1. Komponen Antarmuka Pengguna (*User Interface Component*)

Pengoperasian *chatbot* dimulai saat pengguna menggunakan input teks atau ucapan melalui aplikasi seperti messenger seperti Facebook, Slack, WhatsApp, WeChat, Viber, atau Skype.

2. Komponen Analisis Pesan Pengguna (*User Message Analysis Component*)

User Interface Controller mengarahkan permintaan pengguna ke *User Message Analysis Component* untuk menemukan maksud pengguna dan entitas sesuai dengan *pattern matching* atau *machine learning approaches*. Pesan pengguna bisa berupa teks biasa yang menyimpan semua struktur tata bahasa dan sintaksis dari *input user* atau diproses oleh *Natural Language Processing* (NLP). *Chatbot* harus memahami maksud pengguna dan melakukan tindakan yang diperlukan. *Input* pengguna yang berbeda memicu maksud yang berbeda juga.

3. Komponen Manajemen Dialog (*Dialog Management Component*)

Dialog Management Component mengontrol dan memperbaharui konteks percakapan. Jika *chatbot* tidak dapat mengumpulkan informasi konteks yang diperlukan, *chatbot* akan meminta informasi konteks atau pertanyaan tambahan dari pengguna untuk memenuhi entitas yang hilang. *Dialog Management Component* biasanya mencakup beberapa modul yaitu penanganan ambiguitas (*ambiguity handling*), penanganan data (*data handling*), dan penanganan kesalahan (*error handling*).

4. *Backend*

Setelah *chatbot* mengambil informasi yang dibutuhkan untuk memenuhi kebutuhan pengguna dari *backend* melalui panggilan API eksternal, informasi akan diekstrak dan diteruskan ke *Dialog Management Module* dan *Response Generation*. Dalam *chatbot* berbasis aturan ada basis pengetahuan (*knowledge base*) yang mencakup daftar tanggapan dan berbagai tanggapan terhadap masukan pengguna yang sama untuk menghindari redundansi balasan. *Module Relational DataBase (RDB)* digunakan *chatbot* untuk mengingat percakapan masa lalu dan membuat komunikasi dengan cara yang lebih konsisten dan relevan. Pendekatan ini menghadirkan konsistensi dialog karena memungkinkan *chatbot* mengakses informasi sebelumnya. Basis pengetahuan *chatbot* merupakan hal yang perlu dilakukan, walaupun menuntut dan memakan waktu yang karena dikembangkan secara manual. *Chatbot* berbasis aturan biasanya menyelesaikan basis pengetahuan dengan menanyakan pertanyaan ke pengguna untuk mendorong percakapan yang panjang sehingga membuatnya terlihat seperti manusia.

5. Komponen Penghasil Respon (*Response Generation Component*)

Response Generation Component menghasilkan respon menggunakan satu atau lebih dari tiga model. Model *rule-based* memilih respon dari sekumpulan aturan dengan menghasilkan tanggapan teks baru. Model ini menggunakan basis pengetahuan yang diatur dalam pola percakapan. Model *retrieval-based* lebih fleksibel dalam memilih respon yang sesuai dengan analisis sumber daya yang tersedia menggunakan API. Modul generatif menggunakan *Natural Language Generation (NLG)* untuk merespon *input* bahasa alami, tetapi dibutuhkan kumpulan data dan pelatihan percakapan. Saat *chatbot* menghasilkan respon, *chatbot* menampilkannya kepada pengguna dan menunggu sampai ada umpan balik.

2.2.3 Jurnalisme Percakapan

Menurut Marchionni (2013, p. 253) jurnalisme sebagai percakapan masih belum bisa didefinisikan lebih lanjut karena masih harus dilakukan penulisan dan prediksi fenomena komunikasi yang luas. Namun, Carey dan Anderson (1992), serta Dardenne dan Killenberg (1996) menyimpulkan definisi percakapan sebagai sesuatu yang sosio-psikologis atau interpersonal yang secara khusus dijelaskan bahwa jurnalis dipersepsikan

sebagai manusia sosial dan memiliki kehadiran sosial untuk menyampaikan informasi dengan nada yang santai, informalitas, ramah, dan terbuka dengan kontribusi masyarakat dalam proses berita (Marchionni, 2013, p. 253). Marchionni lebih lanjut menjabarkan studi tentang jurnalisme interaktif dalam Chung (2008), Rosenberry (2005) dan Schultz (2000) serta jurnalisme partisipatif dalam Domingo, dkk (2008) dan Singer, dkk (2011). Marchionni (2013, p. 254) menyarankan agar interaktivitas atau penggunaan strategis alat web oleh jurnalis melibatkan audiens untuk masuk dalam dimensi utama percakapan. Berikut merupakan variabel pengukuran konsep percakapan.

1. *Social presence*: Jurnalis dianggap sebagai manusia
2. *Coorientation*: Kesamaan intelektual yang dirasakan dengan jurnalis
3. *Homophily*: Persepsi kemiripan demografis dengan jurnalis
4. *Friendliness*: Keterbukaan dan aksesibilitas jurnalis untuk berkolaborasi dengan masyarakat
5. *Informality*: Santai dan kasual
6. *Interactivity*: Penggunaan web oleh jurnalis untuk berinteraksi dengan warga.

2.2.4 Efek Komunikasi Massa

Komunikasi massa merupakan komunikasi yang dilakukan melalui media massa dalam bentuk media elektronik (televisi dan radio), media cetak (surat kabar, tabloid, majalah) film, buku, bahkan internet kepada khalayak (Nurudin, 2014, pp. 4-5). Berikut merupakan ciri-ciri komunikasi massa menurut Romli (2016, pp. 4-6).

1. Pesan Bersifat Umum

Pesan yang disampaikan bersifat umum dan terbuka, sehingga tidak ditunjukkan untuk kepada individu atau kelompok tertentu. Pesan komunikasi mengandung peristiwa, fakta, dan opini, tetapi tidak semua dimuat dalam media massa karena harus memenuhi kriteria penting dan menarik.

2. Komunikan Anonim dan Heterogen

Dalam komunikasi massa, komunikator tidak mengenal komunikan karena pesan yang disampaikan menggunakan media, sehingga tidak berinteraksi secara langsung. Khalayak dalam media massa juga bersifat heterogen.

3. Mengutamakan Isi daripada Hubungan

Dimensi isi menunjukkan isi komunikasi, apa yang dikatakan dan yang dilakukan, sedangkan dimensi hubungan menunjukkan bagaimana cara pesan tersebut disampaikan atau bagaimana relasi antarkomunikan.

4. Menimbulkan Keserempakan

Penyebaran pesan kepada khalayak dilakukan secara bersamaan karena sasaran komunikasi yang dicapai relatif banyak.

5. Bersifat Satu Arah

Komunikasi melalui media massa membuat komunikator tidak dapat berinteraksi langsung dengan komunikasi. Komunikasi juga tidak dapat memberikan respon kepada komunikasinya secara langsung.

6. Stimulasi Indera Terbatas

Stimulasi indera dalam media massa bergantung pada jenis media massa, misalnya seperti khalayak hanya bisa mendengarkan siaran radio tanpa melihat peristiwa yang dibicarakan.

7. Umpan Balik Tidak Langsung

Umpan balik atau *feedback* merupakan respon yang diberikan oleh khalayak kepada komunikator. Dalam media massa, komunikator tidak dapat melihat reaksi khalayak secara langsung terhadap pesan yang disampaikan, sehingga khalayak membutuhkan waktu dan perantara lain untuk menyampaikan *feedback* seperti telepon, surel, media sosial, dan lain-lain.

Nurudin (2014, pp. 66-83) menjelaskan fungsi-fungsi komunikasi sejalan dengan perkembangan masyarakat dan teknologi komunikasi dilihat dari perspektif kritis yaitu informasi, hiburan, persuasi, transmisi budaya, pengawasan, korelasi, pewarisan sosial, isan sosial, melawan kekuasaan dan

kekuatan repetitif, menggugat hubungan trikonomi, serta mendorong kohesi sosial. Romli (2016, pp. 14-18) menjelaskan efek komunikasi massa yaitu kognitif, afektif, dan behavioral. Berikut penjelasan mengenai tiap-tiap efek komunikasi massa.

1. Efek Kognitif

Media massa membantu komunikasi untuk meningkatkan kesadaran dalam mempelajari informasi yang berguna, dan mengembangkan keterampilan kognitif. Efek kognitif ini timbul dari diri komunikasi sendiri. Media massa juga memberikan efek profesional kognitif, misalnya ketika kita menonton televisi dengan program ekonomi, hal tersebut membuat kita menjadi lebih mengerti akan investasi dan saham.

2. Efek Afektif

Selain mendapatkan dan mengetahui sebuah informasi, khalayak diharapkan juga dapat merasakannya. Misalnya ketika melihat berita mengenai jatuhnya sebuah pesawat, maka akan timbul rasa iba, kasihan, dan lain-lain. Adapun faktor-faktor yang memengaruhi terjadinya efek afektif dalam komunikasi massa yaitu suasana emosional, skema kognitif, situasi terpaan, dan *predisposisi individual*.

3. Efek Behavioral

Efek behavioral timbul dalam bentuk perubahan perilaku, tindakan, maupun kegiatan yang timbul dari diri khalayak. Misalnya, saat

seorang anak melihat adegan kekerasan dalam tayangan televisi, anak tersebut meniru adegan tersebut. Tidak semua berita yang disampaikan berhasil mengubah perilaku seseorang menjadi lebih baik karena media massa tidak bergantung pada unsur stimulasi.

Dalam pengembangan *chatbot* Relaxa ini, penulis ingin mengetahui apakah konten *chatbot* Relaxa memenuhi efek kognitif dan efek afektif pengguna.

2.2.5 Teknik Pengumpulan Data

Teknik pengumpulan data merupakan teknik yang digunakan untuk mengumpulkan data dengan menggunakan metode-metode tertentu. Menurut Suharsaputra (2014, p. 97), penulis dapat mengombinasikan beberapa teknik pengumpulan data agar lebih meyakinkan. Penulis menggunakan dua teknik pengumpulan data dengan melakukan kuisisioner dan wawancara. Dengan mengombinasikan teknik kuisisioner dengan cara lain, data yang diteliti lebih tinggi keabsahannya karena validitas data survei tergantung pada kejujuran responden itu sendiri (Siyoto & Sodik, 2015, p. 22). Berikut penjelasan mengenai kedua teknik pengumpulan data yang penulis gunakan.

2.2.5.1 Kuisisioner

Kuisisioner atau angket merupakan instrumen penelitian dengan menyebarkan pertanyaan kepada responden untuk dijawab dengan tujuan mendapatkan informasi seperti pendapat, aspirasi, dan lain-lain (Suharsaputra, 2014, p. 95). Kuisisioner dapat dilakukan untuk jumlah responden yang besar maupun yang kecil, sehingga kuisisioner merupakan teknik pengumpulan data yang efisien dengan biaya yang murah. Instrumen kuisisioner yang digunakan penulis adalah skala bertingkat yaitu jawaban responden dilengkapi dengan rentang pernyataan bertingkat (Siyoto & Sodik, 2015, p. 68).

2.2.5.2 Wawancara

Wawancara merupakan pengumpulan data yang dilakukan dengan melakukan percakapan antara responden dengan peneliti untuk mengetahui perasaan, pemikiran, pengalaman, dan hal-hal lainnya yang tidak didapatkan saat melakukan observasi (Suharsaputra, 2014, pp. 213-214). Menurut Paton (dalam Suharsaputra, 2014, pp. 213-214), ada beberapa jenis pertanyaan yang dihadirkan dalam wawancara yaitu,

1. Pertanyaan Pengalaman

Pertanyaan untuk memperjelas beberapa hal yang sebelumnya sudah pernah diamati seperti tindakan, perilaku, dan pengalaman responden.

2. Pertanyaan Pendapat

Pertanyaan untuk mengetahui pendapat responden tentang sebuah peristiwa atau kegiatan tertentu.

3. Pertanyaan Perasaan

Pertanyaan untuk mengetahui perasaan dan pengalaman responden

4. Pertanyaan Pengetahuan

Pertanyaan untuk mengetahui pengetahuan yang dimiliki oleh responden terhadap topik penelitian.

5. Pertanyaan Latar Belakang

Pertanyaan untuk mengetahui demografi responden seperti usia, pendidikan, domisili, dan lain-lain.

Penulis melakukan wawancara semi terstruktur dalam pengumpulan data. Dalam wawancara semi terstruktur, pedoman wawancara atau *interview guide* telah disusun sebelumnya oleh peneliti, tetapi peneliti dapat menanyakan pertanyaan lebih lanjut untuk mengetahui beberapa keterangan lebih rinci (Siyoto & Sodik, 2015, p. 64).

2.2.6 Metode Pengembangan Sistem *Waterfall*

Yurindra mendefinisikan metode *Waterfall* sebagai metode tradisional yang digunakan pada tahun 1960-1970 dan membutuhkan dokumentasi dan struktur di awal pembuatan (2017, p. 41). Model pengembangan ini didasari pada *Systems Development Life Cycle* (SDLC) sehingga sistem harus dilakukan satu per satu secara linier dan sistematis sebelum melanjutkan ke tahap selanjutnya (Yurindra, 2017, p. 34). Hal ini dilakukan untuk meminimalisasi kesalahan sebisa mungkin dan membuat kualitas perangkat lunak tetap terjaga. Namun, pengerjaannya memang membutuhkan waktu yang lama, kaku, dan sulit melakukan perubahan apabila terjadi kekurangan di tahap sebelumnya (Yurindra, 2017, pp. 45-46). Yurindra (2017, pp. 43-45) menjelaskan tahapan dari metode *waterfall* yaitu sebagai berikut.

1. *Requirment Analysis*

Tahap ini merupakan tahap analisis seluruh kebutuhan sistem perangkat lunak yang diinginkan dengan melakukan wawancara, diskusi, survei, dan lain-lain untuk mengetahui kendala, tujuan, dan apa yang diinginkan.

2. *System Design*

Tahap pengembangan ini menghasilkan alur kerja dan detail tampilan perangkat lunak. Hal ini membantu dalam memberikan spesifikasi kebutuhan sistem dan *hardware* dan seluruh arsitektur sistem.

3. *Implementation*

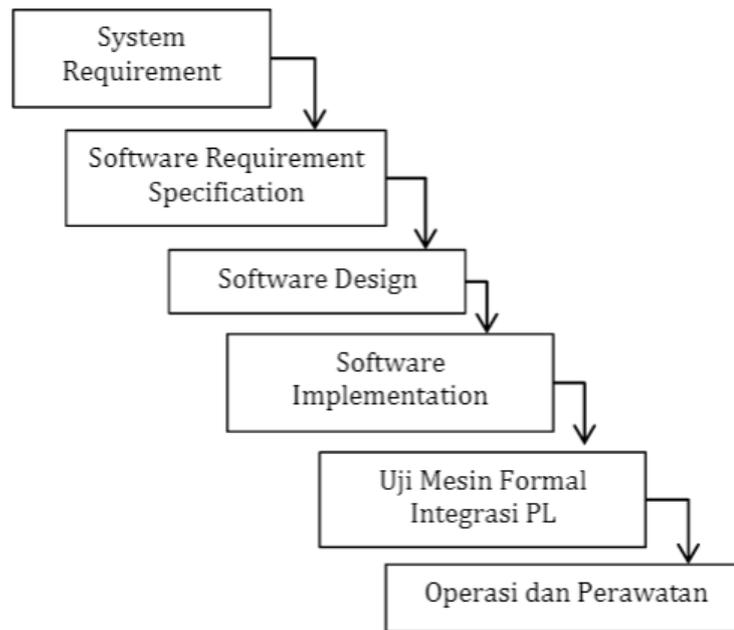
Tahap ini merupakan tahap pemrograman dengan mengimplementasikan kode ke dalam modul-modul yang nantinya akan digabungkan. Modul-modul tersebut akan diperiksa untuk melihat apakah dapat melakukan fungsi yang diinginkan.

4. *Integration and Testing*

Tahap ini dilakukan dengan menggabungkan seluruh modul dan melakukan pengujian keseluruhan sistem untuk mengecek apakah terjadi kesalahan sehingga dapat segera diperbaiki.

5. *Operation and Maintenance*

Perangkat lunak kemudian dijalankan dan diluncurkan ke pengguna. Pemeliharaan juga dilakukan seperti memperbaiki kesalahan atau *bug* yang terjadi setelah perangkat lunak diluncurkan, unit sistem, atau peningkatan sistem.



Sumber: Yurindra, 2020.

Gambar 2.8 Konsep *Waterfall*

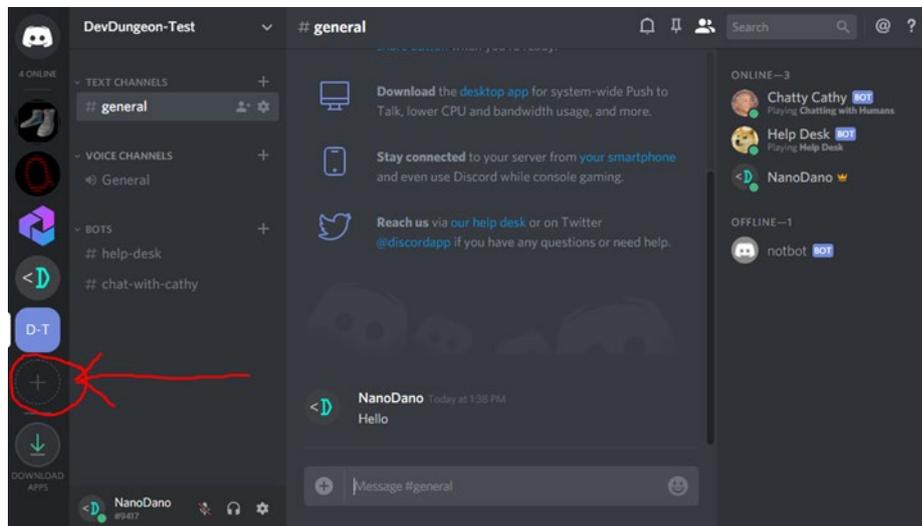
2.2.7 Tahap Pembuatan *Chatbot* Discord

Adamopoulou dan Moussiades (2020, p. 10) mengatakan, tahap pembuatan *chatbot* dimulai dengan perencanaan tujuan, prosedur, dan persyaratan pengguna. Pengembangan *chatbot* dilakukan dengan menggunakan bahasa pemrograman atau platform pengembangan *chatbot*, lalu fungsionalitas *chatbot* diuji secara lokal. *Chatbot* kemudian dipublikasikan di situs laman daring atau pusat data. Pada tahap pembuatan ini, penulis menggabungkan tahap pembuatan menurut Adamopoulou dan Moussiades dan langkah-langkah yang dilakukan oleh DevDungeon (2018), tetapi penulis menyesuaikan cara tersebut dengan cara penulis sendiri

karena aplikasi program yang dimiliki penulis merupakan versi yang paling baru.

1. Membuat *Server* Discord

Untuk membuat *server* Discord, pengguna harus memiliki aplikasi Discord yang dapat diunduh secara gratis dan membuat akun Discord terlebih dahulu. Setelah masuk ke aplikasi, klik pada tanda tambah (+) yang terdapat di sudut kiri bawah yang akan menanyakan apakah pengguna ingin membuat atau bergabung dengan *server*. Klik 'Create My Own', lalu kostumasi *server* dan berikan nama sesuai yang diinginkan, dan klik "Create". *Server* baru akan muncul di kolom kiri.

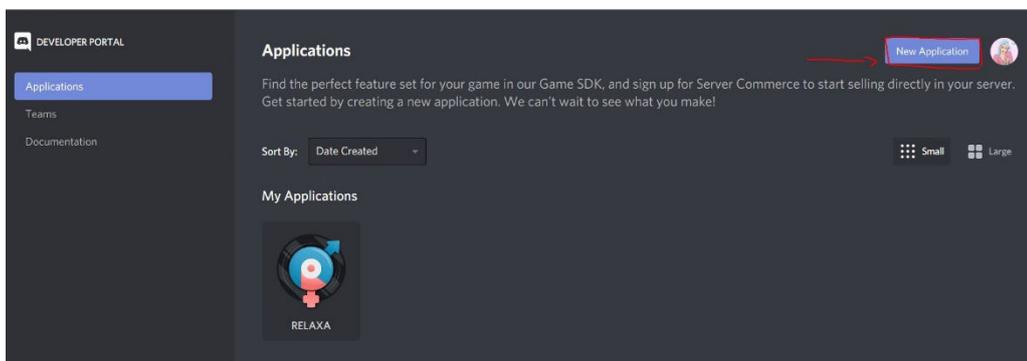


Sumber: devdungeon.com

Gambar 2.9 Cara Membuat *Server* Discord

2. Membuat Aplikasi bot Discord

Untuk membuat aplikasi Discord, pengguna harus mengunjungi laman *Discord Developer Portal* (<https://discord.com/developers/applications>). Klik pada ‘New Application’ di pojok kanan atas, berikan nama sesuai yang diinginkan, dan klik ‘Create’.

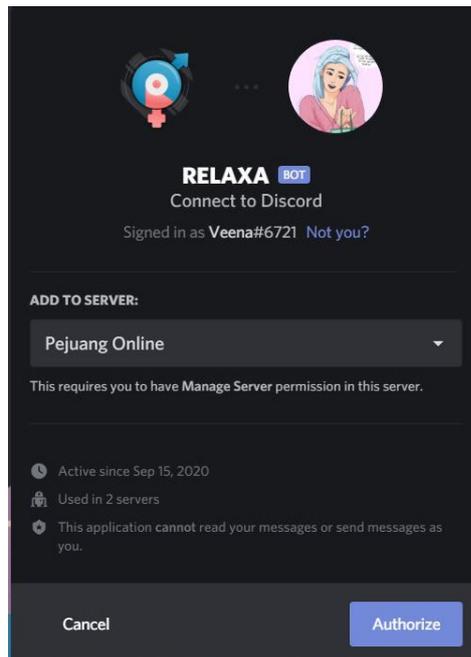


Sumber: Data Pribadi

Gambar 2.10 Laman *Discord Developer Portal*

3. Otorisasi/Undang Bot ke *Server* Discord

Selanjutnya, pengguna akan diarahkan ke halaman arahan atau detail aplikasi. Pada menu ‘General Information’, pengguna dapat mengubah nama aplikasi bot, memberikan deskripsi, mengunggah ikon bot, dan hal umum lainnya. Pada menu ‘OAuth2’, terdapat bagian ‘Scopes’ dan klik pada kotak opsi ‘bot’. Discord akan memberikan tautan untuk mengundang bot ke *server* Discord. Salin alamat tautan tersebut ke *browser* dan pilih *server* mana yang diinginkan, lalu klik ‘Authorize’.



Sumber: Data Pribadi

Gambar 2.11 Izin Autorisasi Saat Mengundang Bot

Pengguna juga dapat mengatur izin bot pada 'Bot Permission' seperti mengirim dan menerima pesan, menambahkan reaksi, serta melakukan hal-hal yang biasa dilakukan oleh pengguna dengan hak istimewa rendah di *server*. Namun, bot tidak dapat melakukan tindakan admin seperti menendang anggota atau mengelola *server*. Untuk melakukan tindakan admin tersebut, pengguna perlu menambahkan izin khusus ke bot serta melakukan langkah-langkah otentikasi tambahan. Pada bagian 'Bot' terdapat token yang dibutuhkan untuk menjalankan bot.

4. Mempersiapkan ruang kerja Node.js

Pertama, penulis harus mengunduh Node.js terlebih dahulu (<https://nodejs.org/en/>). Node.js adalah *open-source* dan *cross-platform environment* untuk mengeksekusi kode JavaScript di luar *browser* (Programming With Mosh, 2018) dan merupakan salah satu *runtime* JavaScript yang populer. Penulis menggunakan Node.js karena gratis, cepat, memiliki dokumentasi *open-source* yang besar, dan basis kode lebih konsisten. (Programming with Mosh, 2018).



Sumber: Programming with Mosh, 2018

Gambar 2.12 Data Keunggulan Aplikasi Node

Penulis juga memerlukan *text editor* yang merupakan suatu program untuk mengedit kode bahasa pemrograman, yaitu Visual Studio Code (<https://code.visualstudio.com/>). Dalam dokumentasinya dijelaskan, Visual Studio Code adalah editor kode yang ringan dan efisien dari Microsoft dengan dukungan *built-in* untuk JavaScript, TypeScript dan Node.js dan memiliki

ekosistem ekstensi yang kaya untuk bahasa lain (seperti C ++, C #, Java, Python, PHP, Go) dan *runtime* (seperti .NET dan Unity). Visual Studio Code juga mendukung operasi pengembangan seperti *debugging*, menjalankan tugas, dan kontrol versi. Selanjutnya, penulis harus membuat folder baru dalam desktop untuk tempat menyimpan pekerjaan pengeditan kode yang dilakukan di dalam Visual Studio Code. Penulis kemudian menambahkan file baru berbentuk .js dengan mengklik ‘New File’.

5. Menginstal modul Discord.js

Untuk menginstal modul Discord.js, klik menu ‘View’ pada Visual Studio Code dan pilih ‘Terminal’ yang akan terbuka di bagian bawah. Dalam terminal ini, penulis menuliskan *npm install discord.js -g* untuk menginstal perpustakaan Discord yang digunakan untuk menyederhanakan interaksi dengan Discord atau kode dan *npm install chalk -g* memungkinkan penulis memiliki warna di terminal (Grzesiek, personal communication, Desember 17, 2020).

6. Menuliskan Tes Sempel kepada Bot

Dalam penulisan kode, penulis menggunakan bahasa pemrograman JavaScript karena mudah dan bisa digunakan dimana saja baik di dalam *browser* maupun di luar seperti aplikasi *mobile*, serta memiliki komunitas yang besar

(Programming with Mosh, 2020 & Telusko, 2020). JavaScript pertama kali diperkenalkan Netscape pada tahun 1996 untuk menghasilkan laman yang menarik (Tolle et al., 2017, p. 190). Menurut Hamedani (Programming with Mosh, 2018), JavaScript adalah salah satu bahasa pemrograman yang sangat populer selama dua dekade dan sangat banyak digunakan di dunia yang saat ini berkembang sangat cepat dibandingkan bahasa pemrograman lainnya.

Hal ini didukung oleh survei yang dilakukan oleh Stack Overflow (dalam Programming with Mosh, 2018) yang menunjukkan bahwa selama tujuh tahun berturut-turut JavaScript menjadi bahasa pemrograman yang populer. Lebih lanjut dijelaskan, JavaScript dulunya hanya digunakan untuk membangun laman interaktif, tetapi saat ini penggunaan JavaScript dapat digunakan untuk membangun *web* atau aplikasi *mobile*, aplikasi *networking real-time*, *command-line tools*, bahkan permainan.



Programming, Scripting, and Markup Languages



Sumber: Programming with Mosh, 2018

Gambar 2.13 Data Popularitas Bahasa Pemrograman, *Scripting*, dan *Markup*

Untuk memulai menjalankan sistem bot, penulis harus mengaktifkan bot tersebut dengan memuat modul Discord dan menghubungkannya. Berikut kode awal yang penulis tuliskan dalam *text editor*.

```
const Discord = require('discord.js');
```

```
const bot = new Discord.Client();
```

Pada baris selanjutnya penulis menambahkan kode untuk men-trigger apabila bot telah diaktifkan. Penulis mengganti variabel *client* dengan *bot* untuk mempermudah.

```
bot.on('ready', () => {
```

```
  console.log(`Logged in as ${bot.user.tag}!`);
```

```
  bot.login('xxxxx');
```

Penulis kemudian harus menghubungkan projek kode ini ke bot. Pada bagian 'xxxxx', masukkan kode token rahasia yang berada di laman *Discord Developer Portal* pada menu 'Bot'.

Klik 'Click to Reveal Token' dan salin kode tersebut. Token ini bersifat rahasia dan dilarang untuk membagikan atau mengunggah token bot karena memungkinkan siapapun untuk masuk ke bot (Tom, 2020, para. 10). Untuk mengaktifkan bot dan menguji apakah bot akan menyala dengan kode yang sudah ditulis, penulis harus menjalankan kode tersebut dengan menuliskan 'node namafile.js' pada terminal. Apabila berhasil, akan terdapat pesan pada terminal yang menandakan bahwa bot sudah aktif (*Logged in as*) dan mengecek di dalam *server* Discord apakah bot sudah *online*.

```
PS D:\Universitas Multimedia Nusantara\COLLE PROJECT> node relaxa.js  
Logged in as COLLE BOT#1889!
```

Sumber: Data Pribadi

Gambar 2.14 Aktivasi Bot yang Berhasil pada Visual Studio Code

7. Uji Coba Bot Secara Lokal

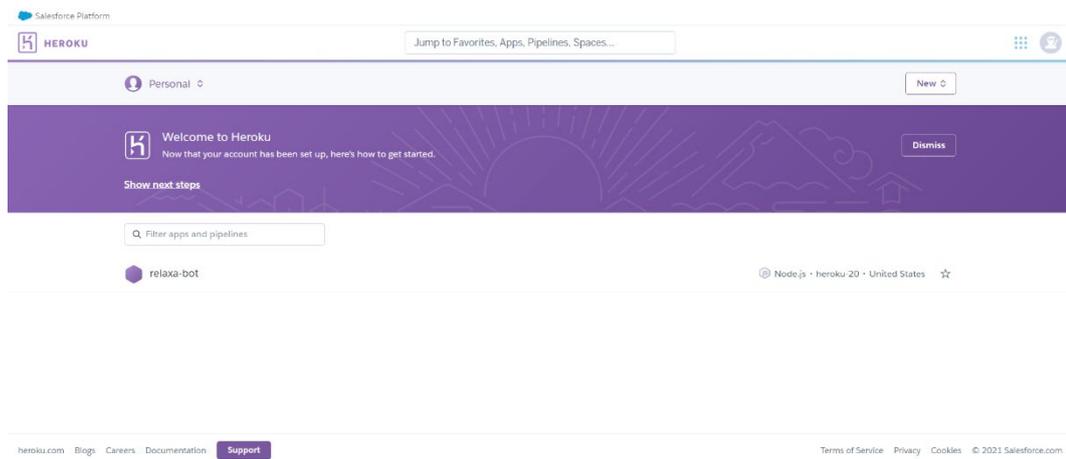
Setelah menuliskan kode dan mengaktifkan bot, penulis menguji coba bot Relaxa di komputer penulis (*local host*) di saluran khusus *server* Discord untuk mengecek apakah semua *command* dapat memanggil respon bot dan memeriksa isi konten apabila terjadi kesalahan penulisan.

8. Mengunggah ke Pusat Data

Selanjutnya, penulis mengunggah file-file bot ke *cloud hosting*. *Cloud hosting* merupakan "tipe *web hosting* yang menggunakan berbagai macam *server* untuk menyeimbangkan

load dan memaksimalkan *uptime*” (Ariata, 2021, para. 1). *Hosting* ini dilakukan agar bot dapat *online* selama 24 jam tanpa mengaktifkan bot lewat *local host*.

Penulis menggunakan GitHub sebagai pusat data dan meng-hosting *chatbot* di Heroku. Heroku adalah platform *cloud* sebagai layanan (PaaS) yang memungkinkan *developer* membangun, mengirimkan, memantau, dan menskalakan aplikasi sepenuhnya di *cloud* (Heroku, n.d.). Heroku memberikan gratis *hosting* selama 550 jam atau sekitar 23 hari.

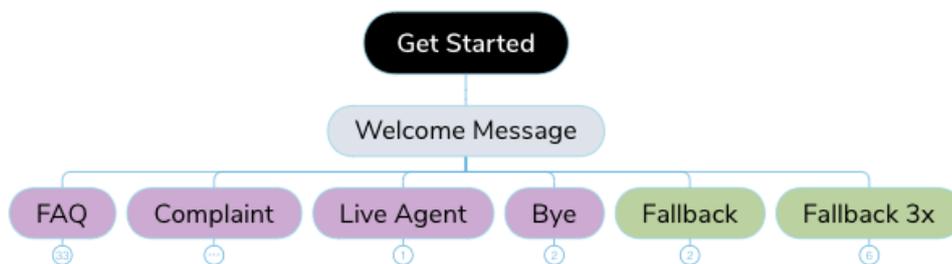


Sumber: Data Pribadi

Gambar 2.16 Tampilan Laman Heroku

2.2.8 Tahap Pembuatan Desain Percakapan

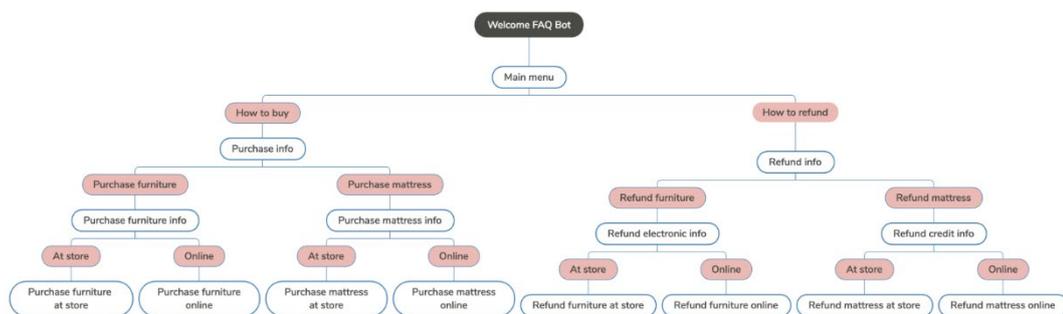
Dalam pembuatan *chatbot* dengan *decision tree-based*, *chatbot flow* harus dibuat terlebih dahulu. Ada beberapa jenis *chat flow* yaitu *floating flow* dan *layered flow*. *Floating flow* biasanya tidak memiliki diagram yang rumit karena hanya terdiri dari pembuka, daftar topik, dan *fallback* (Rembulan, 2020, para. 2).



Sumber: Rembulan, 2020

Gambar 2.17 Contoh *Floating Flow*

Layered flow merupakan aliran diagram seperti pohon yang terdiri dari beberapa lapis keputusan sehingga biasanya *layered flow* dirancang untuk *chatbot* dengan topik dan konten yang lebih beragam (Rembulan, 2020, para. 3).



Sumber: Rembulan, 2020

Gambar 2.18 Contoh *Layered Flow*

Setelah membuat *decision tree*, hal yang selanjutnya dilakukan adalah membuat *decision graph*. *Decision graph* mengarahkan setiap jawaban pertanyaan ke pertanyaan lain sehingga setiap pertanyaan harus secara logis menggabungkan jawaban dari pertanyaan sebelumnya untuk mencari tahu pertanyaan mana yang harus ditanyakan selanjutnya (Lambert, 2018, para. 11). Pertanyaan selanjutnya bergantung pada opsi yang dipilih oleh pengguna. Penulis tidak menemukan formula khusus dalam membuat desain percakapan *chatbot*, tetapi penulis menggunakan pendekatan desain percakapan oleh Black (2018, paras. 12-17) yaitu sebagai berikut.

1. *Outline*

Membuat garis besar gambaran tentang segala sesuatu yang perlu ditulis dengan menguraikan semua alur yang diperlukan untuk mencapai tujuan akhir. Ketahui bahwa alur percakapan memiliki titik masuk dan juga ada terdapat beberapa cara untuk menjangkau satu pertanyaan. Penulis kemungkinan akan menemukan jalur baru yang perlu ditambahkan untuk melanjutkan percakapan dengan cara yang alami.

2. *Flow Map*

Selain garis besar, membuat peta aliran percakapan adalah cara terbaik untuk melihat bagaimana percakapan terhubung. Semakin banyak alur yang dibuat, semakin rumit peta percakapan.

3. *Script*

Setelah Anda memiliki garis besar dan peta, penulis dapat mulai mengembangkan skrip.

4. *Main Flows*

Main flows atau alur utama adalah 1-3 “jalur” yang dapat dilalui pengguna untuk menyelesaikan tujuan bot yang menjadi bagian paling kuat dari skrip. Bagian yang paling kuat ini kemungkinan besar dibagi di beberapa percakapan. Ketika membuat garis besar, arus utama harus berada di bab-bab dari skrip.

5. *Hello, Goodbye*

Hal ini mungkin tampak sederhana, tetapi kesan pertama dari bot sangatlah penting. Penulis perlu membuat pengalaman setelah pengguna selesai mengakses bot.

6. *Catch All State*

Tidak peduli seberapa sempurna skrip yang ditulis, tentu pasti memiliki kelemahan. Pertimbangkan apa yang harus dibalas oleh bot, bagaimana mencoba mengembalikan pengguna ke jalur percakapan, di mana ia akan menempatkan pengguna di dalam alur, dan apa yang terjadi jika bot gagal lebih dari sekali.

Tidak semua pendekatan penulis digunakan dalam implementasi desain percakapan. Penulis hanya menggunakan *outline*, *flow map*, *script*, dan *hello goodbye*. Dalam penulisan skrip atau naskah, penulis mengikuti rumus Tempo mengenai langkah-langkah dalam membuat tulisan (Basuki et al, 2017, pp. 106-112) yaitu,

1. Perencanaan

Mendefinisikan topik dan cakupan isinya, menentukan sudut pandang, membuat garis besar tulisan atau *outline*, rapat redaksi, serta ketajaman memahami data.

2. Pembuatan *Draft*

Memetakan kronologi peristiwa, data yang dibutuhkan, informasi utama lain, membuat *lead*, menulis pokoknya.

3. Perbaikan

Mengevaluasi isi tulisan (apa yang harus dihapus, penggunaan kata dan tanda hubung, terdapat pengulangan kata, kalimat sudah beragam)

4. Penyuntingan

Menajamkan isi berita dan membuat menarik.

5. Perbaikan Kesalahan (*proofreading*)

Merupakan langkah ekstra setelah penyuntingan (menemukan kesalahan yang terlewat pada tahap sebelumnya).

2.2.9 *User Experience Questionnaire (UEQ)*

Pengalaman Pengguna atau *User Experience (EX)* merupakan faktor kunci dalam membangun kualitas produk atau layanan yang didefinisikan oleh ISO yang mencakup mencakup emosi, keyakinan, respon fisik, psikologis, dan lainnya sebagai persepsi dan tanggapan yang dihasilkan dari seorang pengguna terhadap penggunaan produk, sistem, atau layanan (Díaz-Oreiro et al., 2019, p. 1). Standar kuisisioner untuk evaluasi UX berisi serangkaian pertanyaan yang tidak berubah-ubah yang diekspos dalam urutan yang sama menggunakan skala likert atau semantik diferensial untuk mengumpulkan pendapat pengguna tentang karakteristik pragmatis atau hedonistik produk (Díaz-Oreiro et al., 2019, p. 2). Beberapa metode kuisisioner untuk mengevaluasi penggunaan suatu sistem atau produk yang paling dikenal adalah AttrakDiff dengan tujuh poin semantik diferensial, UEQ dengan tujuh poin semantik diferensial, dan meCUE dengan tujuh poin skala likert (Díaz-Oreiro et al., 2019, p. 2).

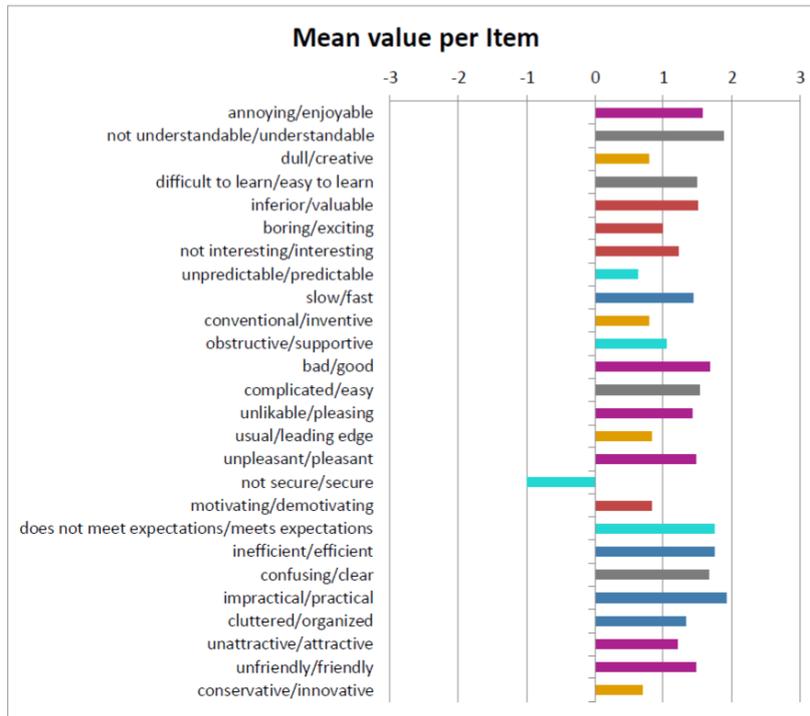


Figure 5: Typical example of an item that is misinterpreted in the given context.

Sumber: Díaz-Oreiro et al., 2019

Gambar 2.19 26 Komponen Pertanyaan UEQ

Hasil penelitian Díaz-Oreiro et al. (2019, p. 6) menunjukkan bahwa 200 dari 553 literatur (36,2%) menggunakan UEQ sebagai standar kuisisioner dan perkembangan penggunaan UEQ melampaui AttrakDiff pada tahun 2017 dan 2018 sebesar 42% dan 47% .

Table 2. Total number of uses of standardized questionnaires

Questionnaire	AttrakDiff	UEQ	meCUE	Total
Uses	341	200	12	553
Percentage	61.6%	36.2%	2.2%	100%

Sumber: Díaz-Oreiro et al., 2019

Gambar 2.20 Jumlah Penggunaan Standar Kuisisioner

Berdasarkan hasil data penelitian geografis Díaz-Oreiro et al. (2019, pp. 7-8), Asia lebih sering menggunakan metode UEQ, terutama di Indonesia (11 dari 33 studi).

Table 3. Total number of uses of standardized questionnaires by geographical region.

Questionnaire	Europe	Asia	North America	South America	Oceania	More than one Region
AttrakDiff	303	7	10	8	5	7
UEQ	150	25	10	7	4	3
meCUE	10	1			1	
Total	463	33	20	15	10	10
Percentage	84.0%	6.0%	3.6%	2.7%	1.8%	1.8%

Sumber: Díaz-Oreiro et al., 2019

Gambar 2.21 Jumlah Penggunaan Standar Kuisisioner Berdasarkan Geografis

Pengukuran dalam UEQ memiliki bentuk semantik diferensial yang setiap komponennya diwakili oleh dua suku dengan arti yang berlawanan (Schrepp, 2019, p. 1). Skala dimulai dengan istilah positif dan separuh lainnya dimulai dengan negatif yang diskalakan dari -3 hingga +3, sehingga -3 mewakili jawaban paling negatif, 0 netral jawaban, dan +3 jawaban paling positif (Schrepp, 2019, pp. 1-2). Contohnya:

attractive o o o o o o *unattractive*

Schrepp et al. (2014, p. 285) menjelaskan struktur skala UEQ dalam enam skala dengan 26 komponen.

1. *Attractiveness*: Kesan keseluruhan produk. Apakah pengguna menyukai atau tidak menyukai produk?

Komponen: *annoying* / *enjoyable*, *good* / *bad*, *unlikable* / *pleasing*, *unpleasant* / *pleasant*, *attractive* / *unattractive*, *friendly* / *unfriendly*.

2. *Perspicuity*: Apakah mudah untuk mengenal produk? Apakah mudah untuk mempelajari caranya menggunakan produk?

Komponen: *not understandable / understandable, easy to learn / difficult to learn, complicated / easy, clear / confusing.*

3. *Efficiency*: Bisakah pengguna menyelesaikan tugas mereka tanpa usaha?

Komponen: *fast / slow, inefficient / efficient, impractical / practical, organized / cluttered.*

4. *Dependability*: Apakah pengguna merasa memegang kontrol terhadap interaksi?

Komponen: *unpredictable / predictable, secure / not secure, meets expectations / does not meet expectations.*

5. *Stimulation*: Apakah menarik dan memotivasi untuk menggunakan produk?

Komponen: *valuable / inferior, boring / exiting, not interesting / interesting, motivating / demotivating.*

6. *Novelty*: Apakah produknya inovatif dan kreatif? Apakah produk menangkap minat pengguna?

Komponen: *creative / dull, inventive / conventional, usual / leading edge, conservative / innovative.*

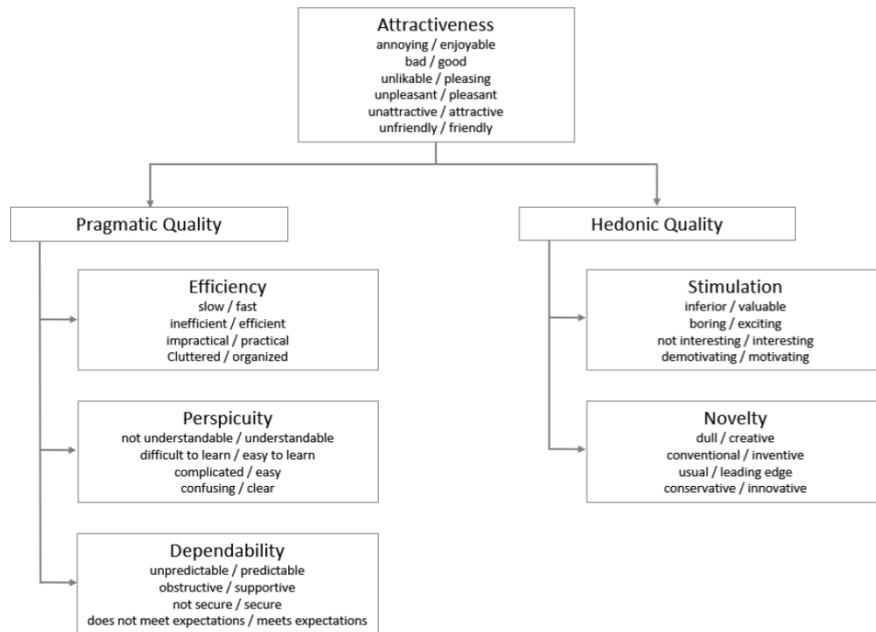


Figure 1: Assumed scale structure of the UEQ.

Sumber: Schrepp, 2019

Gambar 2.22 Struktur Skala UEQ

Schrepp (2019, pp. 2-3) juga menjelaskan tolok ukur dalam UEQ untuk mengklasifikasikan produk menjadi 5 kategori (per skala) yaitu,

1. Luar biasa: Dalam kisaran 10% hasil terbaik.
2. Baik: 10% hasil dalam kumpulan data tolok ukur lebih baik dan 75% dari hasilnya lebih buruk.
3. Di atas rata-rata: 25% hasil di tolak ukur lebih baik daripada hasil untuk produk yang dievaluasi, 50% hasilnya lebih buruk.
4. Di bawah rata-rata: 50% hasil di tolak ukur lebih baik dari hasil untuk produk yang dievaluasi, 25% hasilnya lebih buruk.
5. Buruk: Dalam kisaran 25% hasil terburuk.

Kekurangan dalam menampilkan UEQ lengkap dengan 26 pertanyaan membatasi jumlah pengguna yang bersedia memberikan umpan balik karena terlalu panjang, sehingga dibangun UEQ versi pendek yang tidak memberikan pengukuran pada semua skala UEQ dan hanya terdiri dari delapan komponen (empat komponen mewakili kualitas pragmatis dan empat komponen mewakili kualitas hedonis (Schrepp 2019, p. 11).

The English version of the short UEQ looks like this:

obstructive	o o o o o o o o	supportive
complicated	o o o o o o o o	easy
inefficient	o o o o o o o o	efficient
confusing	o o o o o o o o	clear
boring	o o o o o o o o	exciting
not interesting	o o o o o o o o	interesting
conventional	o o o o o o o o	inventive
usual	o o o o o o o o	leading edge

Sumber: Schrepp, 2019

Gambar 2.23 UEQ Versi Singkat

Namun, versi singkat UEQ-S dimaksudkan hanya untuk skenario khusus atau hanya memberikan pengukuran kasar pada meta tingkat yang lebih tinggi, sehingga tidak memungkinkan untuk mengukur kualitas enam skala UX secara rinci (Schrepp 2019, p. 12).

Table 2
The User Experience Questionnaire items (Indonesian version)

	1	2	3	4	5	6	7		
menyusahkan	<input type="radio"/>	menyenangkan	1						
tak dapat dipahami	<input type="radio"/>	dapat dipahami	2						
kreatif	<input type="radio"/>	monoton	3						
mudah dipelajari	<input type="radio"/>	sulit dipelajari	4						
bermanfaat	<input type="radio"/>	kurang bermanfaat	5						
membosankan	<input type="radio"/>	mengasyikkan	6						
tidak menarik	<input type="radio"/>	menarik	7						
tak dapat diprediksi	<input type="radio"/>	dapat diprediksi	8						
cepat	<input type="radio"/>	lambat	9						
berdaya cipta	<input type="radio"/>	konvensional	10						
menghalangi	<input type="radio"/>	mendukung	11						
baik	<input type="radio"/>	buruk	12						
rumit	<input type="radio"/>	sederhana	13						
tidak disukai	<input type="radio"/>	menggembirakan	14						
lazim	<input type="radio"/>	terdepan	15						
tidak nyaman	<input type="radio"/>	nyaman	16						
aman	<input type="radio"/>	tidak aman	17						
memotivasi	<input type="radio"/>	tidak memotivasi	18						
memenuhi ekspektasi	<input type="radio"/>	tidak memenuhi ekspektasi	19						
tidak efisien	<input type="radio"/>	efisien	20						
jelas	<input type="radio"/>	membingungkan	21						
tidak praktis	<input type="radio"/>	praktis	22						
terorganisasi	<input type="radio"/>	berantakan	23						
atraktif	<input type="radio"/>	tidak atraktif	24						
ramah pengguna	<input type="radio"/>	tidak ramah pengguna	25						
konservatif	<input type="radio"/>	inovatif	26						

Sumber: Santoso et al., 2016

Gambar 2.24 UEQ Versi Lengkap Bahasa Indonesia

2.2.10 Skala Likert

Menurut Yusuf (2014, p. 222), skala merupakan salah satu teknik yang digunakan untuk mengumpulkan data karena mempunyai validitas yang tinggi, reliabilitas yang andal, dan utilitas yang baik. Lebih lanjut, Yusuf menjelaskan langkah-langkah dalam penyusunan skala yaitu menentukan studi literatur dengan menentukan indikator, spesifikasi, dan dimensi objek penelitian serta menyusun indikator yang sesuai dengan aspek-aspek yang diukur (2014, p. 222).

Menurut Sugiyono (2013, p. 93), skala likert merupakan pengukuran dalam penelitian yang mengukur pendapat, sikap, persepsi, dan pendapat individu atau kelompok terhadap suatu fenomena sosial. Skala Likert merupakan skala yang dikembangkan oleh Rensis Likert untuk mengukur sikap individu terhadap suatu dimensi ke arah kontinuitas dari setiap butir soal (Yusuf, 2014, p. 222). Peneliti harus merumuskan pertanyaan mengenai topik tertentu. Pernyataan positif dan negatif diusahakan seimbang jumlahnya dengan penjelasan setiap instrumen yang mudah dibaca, dipahami, dan mudah diisi oleh responden, sehingga penulis harus menentukan mana pernyataan yang bersifat positif dan negatif (Yusuf, 2014, pp. 222-224). Skala Likert memiliki skor dengan lima poin (1, 2, 3, 4, dan 5) dan tujuh poin (1, 2, 3, 4, 5, 6, dan 7). Skala Likert dengan pemberian skor lima poin menggunakan nilai sangat setuju, setuju, ragu-ragu/netral, tidak setuju, dan sangat tidak setuju (Sugiyono, 2012, p. 93). Untuk menjaga konsistensi pengukuran sikap, bobot jawaban harus disusun

terbalik untuk pernyataan negatif. Apabila responden memilih pernyataan yang positif, pilihan jawaban sangat setuju akan diberikan poin 5, sedangkan untuk pernyataan negatif, jawaban sangat tidak setuju akan diberikan 5 poin (Yusuf, 2014, p. 224).

2.2.11 Simbol Seksualitas

Simbol yang digunakan untuk mendefinisikan jenis kelamin biologis dituliskan dalam jurnal *Taxon* (1962) yang berjudul “The Origin of Male and Female Symbols of Biology” oleh William T. Stearn (Sulaeman, 2017, para. 8). Carl Linnaeus merupakan orang pertama yang menggunakan tanda ini dalam disertasi *plantae hybridae* (1751) sebagai konteks biologis yaitu Venus (Unicode: U+2640) yang berbentuk lingkaran dengan tanda seperti salib mengarah ke selatan yang merujuk ke tanaman betina dan Mars (Unicode: U+2642) yang berbentuk lingkaran dengan tanda panah mengarah ke timur laut yang merujuk tanaman laki-laki (Sulaeman, 2017, paras. 9-10 & Today I Found Out, 2017).

Penggunaan lambang ini kemudian berkembang lebih lanjut dalam bidang zoologi, biologi, hingga genetika modern. Menurut Ernest Cassirer (dalam Sulaeman 2017, paras. 15-17), simbol laki-laki tersebut menggambarkan aktivitas berburu dengan tombak yang menandakan maskulinitas sedangkan simbol perempuan menunjukkan seperti seorang wanita yang bercermin.



Sumber: pngall.com

Gambar 2.21 Simbol Venus dan Mars

Penulis menggunakan dua karakter yaitu karakter remaja perempuan yang ditandai dengan simbol Venus dan karakter remaja laki-laki dengan simbol Mars. Nama karakter tersebut juga diambil dari peleburan nama Venus menjadi Vena untuk karakter remaja perempuan dan Mars menjadi Mark untuk karakter remaja laki-laki.



Sumber: Data Pribadi

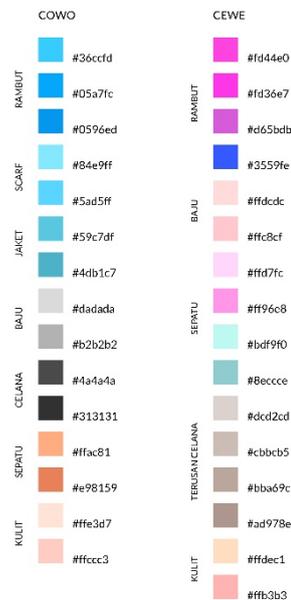
Gambar 2.22 Karakter Vena (kiri) dan Mark (kanan)

2.2.12 Simbolisme Warna

Penulis menggunakan referensi karakter dan logo dengan dasar warna merah muda untuk perempuan dan biru untuk laki-laki. Referensi warna simbolik mendeskripsikan karakter serta perilaku manusia, bersifat simbolis mewakili mental, konseptual, kualitas, dan merupakan budaya sehingga referensi warna yang digunakan tidak sama di seluruh dunia (Lauer & Pentak, 2011, p. 288). Pemilihan warna ini diambil berdasarkan sejarah mengapa perempuan identik dengan merah muda dan laki-laki dengan biru. Namun, penulis menekankan bahwa perbedaan warna ini tidak

bertujuan untuk membatasi warna yang digunakan oleh perempuan dan laki-laki bahkan memberi stereotip warna tertentu dengan peran gender.

WARNA KARAKTER



Sumber: Data Pribadi

Gambar 2.23 *Color palette* untuk karakter Vena dan Mark

Menurut Paoletti (dalam Hartmann, 2011, paras. 2-3) pakaian bayi yang berwarna pastel tersebut awalnya tidak spesifik menunjuk gender tertentu berdasarkan artikel Ladies 'Home Journal' bulan Juni 1918 yang menyebutkan:

"Aturan yang berlaku umum adalah merah muda untuk anak laki-laki, dan biru untuk anak perempuan. Alasannya adalah karena merah muda menjadi warna yang lebih tegas dan lebih kuat, lebih cocok untuk anak laki-laki, sedangkan biru, yang lebih halus dan halus, lebih cantik untuk gadis itu."

Pada tahun 1940-an pabrik memilih warna merah muda untuk anak perempuan dan biru untuk anak laki-laki, sehingga *Baby Boomers* dibesarkan dengan memakai dua warna yaitu warna merah muda dan biru yang kemudian tren ini muncul kembali di pertengahan 80-an dengan perkembangan pengujian prenatal (Hartmann, 2011, para. 5). Penggunaan simbol seksualitas dan warna biru serta merah muda digunakan dalam logo atau ikon bot Relaxa dengan bagian tengah menyerupai bentuk huruf R yang berarti Relaxa. Lingkaran hitam berbentuk seperti piringan hitam digunakan untuk membuat logo menjadi lebih estetik dan *compact*. Selain itu lingkaran piringan hitam juga merupakan penyatuan harmonisasi antara laki-laki dan perempuan.



Sumber: Data Pribadi

Gambar 2.24 Logo Bot Relaxa