

# Chatbot: Reservasi Restoran *Online* pada *Facebook Messenger*

Saiqul Haq<sup>1</sup>, Bayu Waspodo<sup>2</sup>, Nuryasin Nuryasin<sup>3</sup>

**Abstrak**—Akses *chatbot* yang diberikan oleh *Facebook* ke dalam aplikasi *Messenger* memberikan peluang besar bagi pelaku bisnis untuk meraih pengguna *Facebook* yang mencapai 1,2 miliar. Sebagai perusahaan yang pelanggan utamanya berasal dari *Facebook*, maka AppServation Co., Ltd perlu melakukan pengembangan pada produknya untuk memberikan layanan secara langsung kepada pelanggannya melalui *chatbot*. Penelitian ini bertujuan meningkatkan layanan pelanggan AppServation Co., Ltd dengan menambahkan fitur reservasi berupa *chatbot* yang terdapat pada *Facebook Messenger*. Reservasi dapat dilakukan tanpa mengharuskan pelanggan mengunjungi halaman reservasi yang berada di luar *Facebook*. Metode pengembangan sistem yang digunakan untuk membangun *chatbot* pada sistem berjalan adalah menggunakan metode pendekatan analisis terstruktur (*waterfall*). Pengujian atas sistem yang dibangun dengan menggunakan metode *behavior-driven development*. Hasil pengujian sistem menunjukkan bahwa dengan sistem *chatbot* mampu melayani permintaan reservasi pada platform *Facebook Messenger*.

**Kata Kunci**—*Chatbot*, *Rule-based model*, *Facebook Messenger*, *Waterfall*, *Behavior-driven Development*

## I. PENDAHULUAN

Sebagian teknologi saat ini turut memengaruhi proses pemesanan tempat pada restoran-restoran yang awalnya melalui telepon kini bisa dilakukan secara daring dan bisa dilakukan kapan saja dan bahkan memudahkan bagi pemilik restoran untuk manajemen reservasi pada restorannya [1]. AppServation Co., Ltd adalah perusahaan teknologi yang menyediakan layanan reservasi restoran secara daring. Sistem reservasi tersebut bernama *Hungry Hub*, yang bisa diakses oleh pelanggan dalam bentuk aplikasi *web* dan *mobile*. Adapun fitur-fitur yang tersedia pada *Hungry Hub* adalah mengenai reservasi tempat, pengaturan ketersediaan tempat, katalog menu, reward dan lain-lain. Berdasarkan data reservasi bulanan yang dilaporkan oleh sistem, ternyata reservasi terbanyak dilakukan oleh pengguna *Facebook*. Berdasarkan hasil wawancara yang dilakukan, pengguna *Facebook* tersebut harus melakukan

pemesanan melalui formulir yang bisa diakses pada salah satu halaman di situs *Hungry Hub*. Hal tersebut berakibat pada pelanggan harus meninggalkan aplikasi atau situs *Facebook*. Pada bulan April 2016, *Facebook* memberikan akses bagi pihak ketiga untuk berkomunikasi dengan pengguna *Facebook* secara langsung melalui teknologi *chatbot*. Hal tersebut memberikan peluang bagi sistem mana pun, termasuk bagi perusahaan AppServation.

*Chatbot* adalah aplikasi komputer yang berinteraksi dengan pengguna menggunakan *natural languages*. Istilah *chatbot* telah digunakan untuk beberapa jenis program seperti *machine conversation system*, *virtual agent*, *dialogue system*, dan *chatterbot*. Arsitektur sistem *chatbot* mengintegrasikan sebuah *language model* dan *computational algorithmic* untuk mengemulasikan komunikasi informal antara manusia dan komputer menggunakan bahasa alami [2]. Penggunaan *chatbot* banyak ditemukan pada kehidupan sehari-hari, seperti *help desk tools*, *automatic telephone answering systems*, *business* dan *e-commerce*. Pemanfaatan *chatbot* pada restoran digunakan untuk mengerjakan beberapa peran yang biasanya dilakukan oleh staf restoran, seperti melayani pemesanan meja, menjawab pertanyaan seputar menu yang tersedia, melakukan promosi, dan lain-lain. Selain itu, *chatbot* mampu bekerja selama 24 jam setiap harinya [3]. Penggunaan *chatbot* pada sistem reservasi online telah dibuktikan manfaatnya oleh beberapa penelitian yang telah dilakukan sebelumnya. Penelitian yang dilakukan oleh [4] yang menerapkan *chatbot* pada situs hotel menunjukkan bahwa *chatbot* berfungsi sebagai alat informasi dan menambah persentase *booking online*. Dari analisis atas hasil pada penelitian yang serupa menunjukkan bahwa waktu proses pemesanan menjadi lebih efisien karena berada dalam satu *interface* [5].

Penelitian-penelitian mengenai *chatbot* juga telah dilakukan sebelumnya, namun belum ada yang menggunakan platform *Facebook Messenger*, misalnya seperti penggunaan *chatbot* untuk *e-commerce* yang berjalan pada platform *Prestashop* [6], lalu penelitian penggunaan *chatbot* untuk media informasi yang berjalan pada sistem internal situs Dinas Kebudayaan dan

<sup>2</sup>B. Waspodo, Prodi Sistem Informasi UIN Syarif Hidayatullah Jakarta, Indonesia (e-mail bayu: [waspodo@uinjkt.ac.id](mailto:waspodo@uinjkt.ac.id))

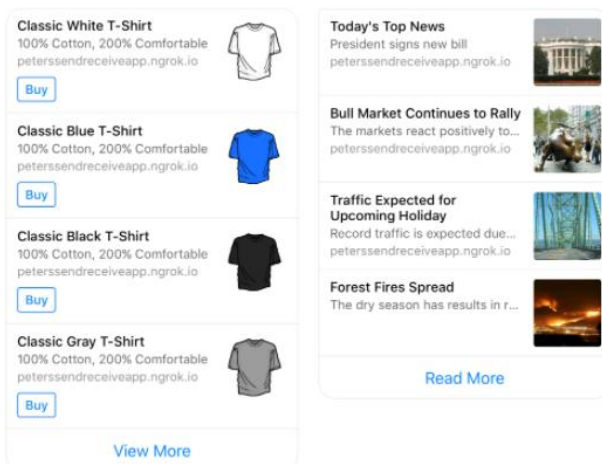
<sup>3</sup>N. Nuryasin, Prodi Sistem Informasi UIN Syarif Hidayatullah Jakarta, Indonesia (e-mail: [nuryasin@uinjkt.ac.id](mailto:nuryasin@uinjkt.ac.id))

Pariwisata Kota Bandung [7], begitu pula dengan penelitian *chatbot* untuk pemesanan hotel yang berjalan pada sistem internalnya sendiri [5]. Penelitian ini mengungkapkan bahwa *chatbot* yang berjalan pada *Facebook Messenger* dan bisa membantu pengguna Facebook agar bisa melakukan reservasi pada sistem *Hungry Hub* tanpa perlu meninggalkan laman atau aplikasi *Facebook*.

## II. KAJIAN PUSTAKA

*Chatbot* adalah perangkat lunak komputer yang mampu berinteraksi dengan *users* menggunakan bahasa alami, seperti memberikan pesan kepada *chatbot* dengan kalimat “Bagaimana kabar cuaca hari ini?”, lalu *chatbot* akan menerjemahkan pesan tersebut, dan menangkap isi pesan dengan teknik-teknik tertentu, hingga pada akhirnya *chatbot* menjawab bahwa cuaca hari ini adalah terik, hujan, ataupun mendung berdasarkan data cuaca yang didapatkan dari sumber tertentu. Terdapat beberapa istilah yang telah digunakan untuk sebuah *chatbot*, seperti *machine conversation system*, *virtual agent*, *dialogue system*, dan *chatterbot* [8]. Untuk memahami pesan yang diberikan oleh pengguna, ada beberapa jenis arsitektur yang bisa digunakan, misalnya seperti *rule-based architecture* dan *corpus-based architecture*. Pada penelitian ini, *chatbot* dibangun menggunakan *rule-based architecture*. Sebagai sebuah sistem, umumnya *chatbots* terdiri atas beberapa komponen, seperti *input analyzer*, *dialog manager*, *specific agent*, dan *answer generator* [9], [10].

*Facebook Messenger* adalah layanan *instant messaging* dan aplikasi perangkat lunak yang terintegrasi dengan fitur *chatting* yang dimiliki *Facebook* [11], sehingga melalui aplikasi tersebut, pengguna bisa melakukan *chatting* melalui 2 cara, cara yang pertama adalah melalui situs *Facebook* yang bisa diakses pada <https://facebook.com>, dan yang kedua melalui situs *Facebook Messenger* yang bisa diakses pada <https://messenger.com>. Selain berbasis *web*, *Facebook Messenger* juga tersedia dalam aplikasi *mobile* berbasis *Android*, *iOS*, *Windows Phone*, *Tizen*, dan *Blackberry*, sehingga jangkauan penggunaannya sangat luas sekali. Untuk

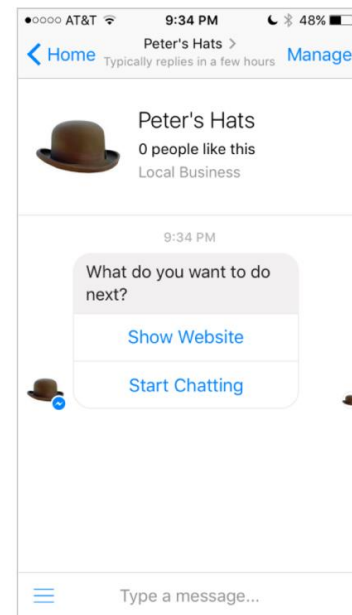


Gambar 1. Contoh *structure template* dalam bentuk *list*

memahami pesan yang diberikan oleh pengguna, ada beberapa jenis arsitektur yang bisa digunakan, misalnya seperti *rule-based architecture* dan *corpus-based architecture*. Pada

penelitian ini, *chatbot* dibangun menggunakan *rule-based architecture*. Sebagai sebuah sistem, umumnya *chatbots* terdiri atas beberapa komponen, seperti *input analyzer*, *dialog manager*, *specific agent*, dan *answer generator* [12].

Fitur *bot* pada *Facebook Messenger* memiliki kemampuan untuk mengirim pesan dalam format teks, suara, video, gambar, dan berkas. Selain itu terdapat pula fitur yang disebut dengan *structured templates*, yaitu fitur yang menyediakan kemampuan untuk mengirim pesan dalam bentuk yang terstruktur, misalnya seperti yang ditunjukkan pada Gambar 1 dan 2. Selain untuk saling mengirim pesan, terdapat pula fitur. *Bot* yang menyediakan kemampuan untuk mengotomasi balasan atas pesan yang dikirim oleh pengguna. Pesan yang



Gambar 2. Contoh *Structured Template* dalam Bentuk Tombol

dikirim oleh pengguna, dikirim ke *server* yang telah dibuat oleh *developer* melalui *webhook*. Sebuah *webhook* mengirimkan data ke aplikasi lain secara *real-time*, yang berarti Anda mendapatkan data dengan segera.

## III. METODE PENELITIAN

Metode pengumpulan data pada penelitian ini dilakukan dengan melakukan observasi dan wawancara. Observasi dilakukan dengan cara menganalisis proses-proses yang terdapat pada aplikasi-aplikasi *Hungry Hub*, memantau *log* aplikasi dan memeriksa laporan-laporan yang ada. Observasi ditujukan untuk mendapatkan bagaimana proses dan mekanisme sistem yang sedang berjalan dan masalah-masalah teknis apa yang sedang terjadi pada sistem. Wawancara dilakukan dengan mewawancarai beberapa pihak, seperti kepada CEO, dan CTO perusahaan AppServation Co, Ltd untuk mendapatkan data dan informasi yang dibutuhkan. Adapun metode pengembangan sistemnya adalah menggunakan metode *structured analysis* atau *waterfall*, namun tanpa melakukan fase *maintenance*. Metode ini diawali dengan melakukan

perencanaan, yaitu mengumpulkan data profil bisnis perusahaan, menganalisis proses berjalan pada area sistem yang diteliti, menganalisis kondisi area sistem saat ini yang sedang diteliti menggunakan metode analisis *SWOT* dan menggali masalah yang terjadi dengan cara observasi, menentukan ruang lingkup dan *constraint* atas sistem yang diusulkan

Fase selanjutnya adalah dengan melakukan analisis sistem, yaitu dengan melakukan pemodelan kebutuhan dan objek. Dengan melakukan dua hal tersebut, maka dihasilkan strategi pengembangan sistem yang akan dilakukan. Pemodelan dilakukan dengan menganalisis kebutuhan-kebutuhan apa saja pada sistem saat ini, kemudian membuat daftar periksa kebutuhan-kebutuhan apa saja untuk sistem usulan. Aktivitas pemodelan objek dibagi menjadi tiga bagian, yaitu: pemodelan fungsional, struktural, dan perilaku. Analisis pemodelan objek dilakukan dengan menggunakan teknik-teknik UML seperti *use case diagram*, *class diagram*, *sequence diagram*, *state transition diagram*, dan *activity diagram*.

Setelah melakukan analisis sistem, kemudian fase selanjutnya adalah fase perancangan, yaitu dengan merancang data dan arsitektur sistem. Perancangan data dilakukan dengan menentukan entitas terlebih dahulu, elemen-elemen, normalisasi struktur *database* dan yang terakhir adalah membuat kamus data. Adapun fase terakhir adalah fase implementasi, yaitu *coding* atau pemrograman yang dilanjutkan dengan pengujian *Chatbot* dengan *behavior-driven development*.

#### IV. HASIL DAN PEMBAHASAN

Bagian ini menjelaskan hasil dari masing-masing fase pada metode pengembangan sistem yang telah dilakukan.

##### A. Perencanaan Sistem

AppServation Co., Ltd adalah perusahaan yang berlokasi di Thailand dan menyediakan layanan sistem reservasi untuk restoran dengan nama *Hungry Hub*. *Hungry Hub* ditemukan di bawah angan-angan bahwa konsumen-konsumen pada saat itu menginginkan kemampuan untuk mencari restoran-restoran lokal berdasarkan lokasi atau jenis masakan dan kemudian melakukan pemesanan tempat pada semua restoran yang diinginkan hanya dengan menekan satu tombol pada komputer atau *smartphone*.

Pada realitasnya restoran-restoran tidaklah dibuka selama 24 jam dalam sehari dan 7 hari dalam seminggu, sehingga membatasi kemampuan restoran untuk menerima reservasi ketika calon konsumen mereka berencana untuk memesan kepadanya. Banyak restoran menggunakan cara kuno seperti menyimpan dan merekam data konsumennya secara manual. Dengan menggunakan sistem *Hungry Hub*, reservasi bisa dilakukan dan direkam secara elektronik. Hal tersebut lebih tepat dan mampu mengurangi celah kesalahan. Saat ini perusahaan AppServation Co., Ltd hanya menyediakan sistem reservasi restoran saja. Adapun fitur-fitur yang terdapat pada sistem tersebut terbagi dalam dua perspektif berikut:

##### 1) Pemilik restoran:

Sebagai pemilik restoran, ia bisa melakukan pengelolaan data profil, reservasi, pelanggan, katalog menu dan jadwal beserta berapa kursi yang tersedia. Selain itu sistem juga menyajikan laporan reservasi per-harinya.

##### 2) Pelanggan:

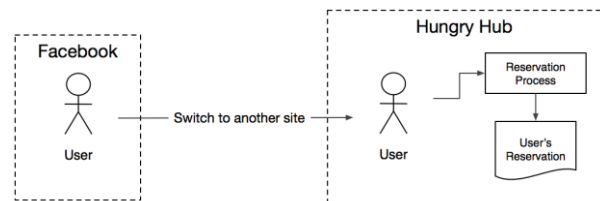
Bagi pelanggan yang sudah maupun belum terdaftar, keduanya memiliki akses untuk melakukan permintaan, perubahan jadwal, dan pembatalan reservasi. Selain itu mereka juga bisa melakukan review atas pengalaman yang dialaminya ketika sedang di restoran. Namun bagi pelanggan yang terdaftar, ia akan mendapatkan *reward* jika ia hadir di restoran sesuai dengan jadwal reservasi. Pelanggan terdaftar juga bisa melihat katalog-katalog menu pada aplikasi *mobile*. Ada tiga jenis produk perusahaan AppServation Co., Ltd yang bisa diakses bagi pengguna, yaitu:

##### 3) Aplikasi berbasis web yang berfungsi hanya untuk membuat reservasi tanpa melakukan pendaftaran pengguna terlebih dahulu.

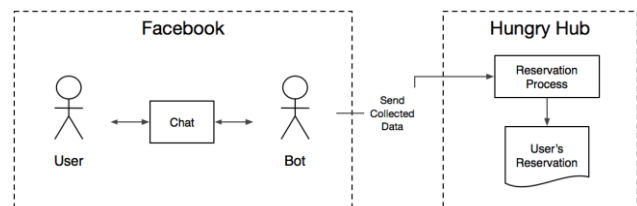
##### 4) Aplikasi berbasis mobile yang memiliki banyak fitur, namun pengguna harus melakukan pendaftaran terlebih dahulu.

##### 5) Aplikasi berbasis web dan mobile yang diperuntukkan bagi pemilik restoran untuk mengelola restorannya.

Ada tiga jenis pelanggan pada perusahaan AppServation Co., Ltd, yaitu pemilik restoran, pengguna terdaftar, dan pengguna yang tidak terdaftar. Adapun yang dimaksud dengan pengguna tidak terdaftar adalah pelanggan yang tidak memiliki akun khusus pada aplikasi *Hungry Hub*, sehingga ia harus memberikan data pribadi setiap kali melakukan reservasi. Ada banyak kompetitor perusahaan AppServation Co., Ltd, yaitu *Eatigo.com*, *Chope.co*, *Opentable.com*, *Urbanspoon.com*, *Zomato.com*, *Chatobook.com* dan *Orderbot.co*. Dari beberapa kompetitor yang disebutkan, hanya *Chatobook.com* dan *Orderbot.co* saja yang sistemnya tersedia dalam bentuk *chatbot*.



Gambar 1. Alur Reservasi pada Sistem Saat Ini Jika Diakses dari Facebook



Gambar 2. Alur Reservasi pada Sistem Usulan Jika Diakses dari Facebook

Hasil studi kelayakan operasional menunjukkan pengguna yang terdapat di Facebook bisa berkomunikasi dengan sistem pihak ketiga tanpa perlu meninggalkan laman *Facebook*. Sebagaimana ditunjukkan oleh Gambar 3 dan 4.

**B. Analisis Sistem**

Pada tahap ini, analisis terhadap *form booking* berbasis *web* yang berjalan saat ini dilakukan untuk mengidentifikasi kebutuhan-kebutuhan sistem usulan seperti keluaran, masukan, kontrol dan prosesnya.

Hasil identifikasi menunjukkan tiga proses bisnis yang ada, yaitu permintaan reservasi, permintaan bantuan reservasi, dan penautan *link* ke laman *Facebook*. Maksud dari permintaan bantuan reservasi adalah permintaan bantuan kepada *staff AppServation* untuk memberitahu pelanggan ketika ada kursi yang tersedia pada tanggal dan waktu tertentu di restoran yang diinginkan.

Proses selanjutnya adalah melakukan analisis kebutuhan pada sistem usulan. Adapun yang dibutuhkan pada sistem yang diusulkan adalah sebagai berikut:

1) *Proses Otorisasi Chatbot*: Pemilik restoran harus memberikan access token halaman Facebook Page restorannya

2) *Proses Permintaan Reservasi*:

- a. Masukan: *Bot* harus menanyakan data kapan tanggal, waktu, dan berapa jumlah orang dewasa dan anak kecil yang memesan. *Bot* harus menanyakan data pribadi pemesan, yaitu nama, nomor telepon, dan email. *Bot* harus menanyakan apakah pelanggan memiliki permintaan khusus, misalnya seperti permintaan kursi bayi, lokasi di pojok atau di luar, dan sebagainya. *Bot* harus menanyakan data kartu kredit pelanggan jika pihak restoran memerlukan konfirmasi reservasi yang dilakukan. Data tersebut adalah nama pemilik, nomor identitas, tanggal kadaluwarsa, dan nomor CVV kartu kredit. *Bot* harus mengkonfirmasi semua data yang diberikan oleh pelanggan. *Bot* harus mengkonfirmasi data terekam yang belum di *submit* dalam waktu 24 jam.
- b. Proses: *Bot* harus menerjemahkan data tanggal, dan waktu yang diberikan pelanggan ke dalam format yang *valid* bagi sistem. *Bot* harus memvalidasi nomor telepon dan email yang diberikan oleh pelanggan. *Bot* harus melakukan *Authorized Payment Credentials* atas data kartu kredit yang diberikan oleh pelanggan. *Bot* harus mengirim data reservasi ke *server Hungry Hub*. *Server* harus menangkap data reservasi yang dikirim oleh *bot* dan menyimpannya ke *database*.
- c. Keluaran: Sistem reservasi harus memproduksi nomor identitas reservasi yang dibuat oleh pelanggan.
- d. Kontrol: Jika pelanggan memiliki reservasi di waktu yang akan datang, maka bot hanya melayani permintaan reservasi jika waktu reservasi beda satu jam dari waktu reservasi yang akan datang tersebut

3) *Proses Permintaan Bantuan Reservasi*

- a. Masukan: *Bot* harus menanyakan data nama, email, nomor telepon, isi pesan bantuan, jumlah orang pemesan, tanggal, dan jam reservasi yang diinginkan. *Bot* harus mengkonfirmasi data terekam yang belum diserahkan (*submit*) dalam waktu 24 jam.
- b. Proses: *Bot* harus memvalidasi nomor telepon dan email yang diberikan pelanggan. *Bot* harus menerjemahkan

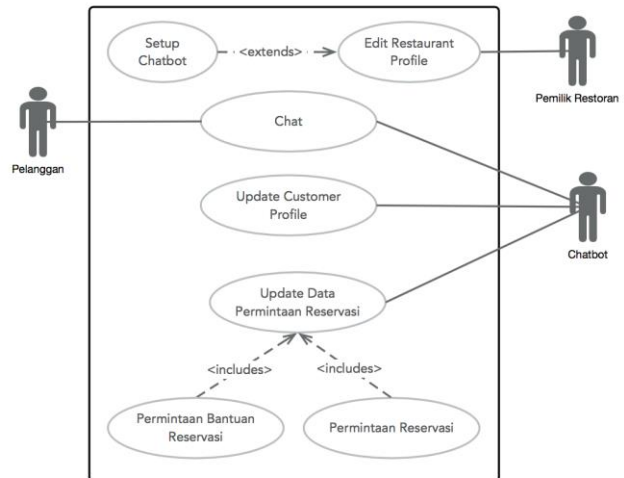
data tanggal dan jam reservasi yang diberikan pelanggan dalam format yang dimengerti sistem. *Bot* harus memvalidasi data tanggal dan jam reservasi yang diberikan pelanggan. *Bot* harus mengirim data yang diberikan oleh pelanggan ke server. *Server* harus menangkap data yang dikirim oleh *bot* dan menyimpannya ke basis data.

- c. Keluaran: Sistem harus menghasilkan laporan bahwa ada pelanggan yang meminta bantuan reservasi kepada staf *AppServation*.

Proses selanjutnya dalam fase analisis sistem adalah melakukan pemodelan fungsional, yaitu dengan menganalisis aktor dan identifikasi *use case*, seperti pada Tabel 1 dan gambar 5.

Tabel 1.  
Hasil Identifikasi Aktor

Symbol	Quantity	Conversion from Gaussian and CGS EMU to SI <sup>a</sup>
1	Pelanggan	Pengguna <i>Facebook</i> yang melakukan permintaan reservasi dan permintaan bantuan reservasi.
2	Pemilik Restoran	Memberikan akses bagi chatbot untuk mengakses laman <i>Facebook Page</i> restorannya.
3	Chatbot	Berinteraksi dan menjawab permintaan pelanggan.



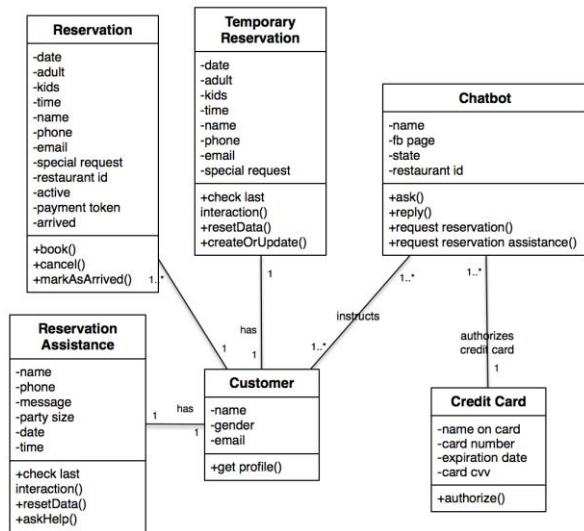
Gambar 3. Hasil Identifikasi Use Case



Tabel 2.  
 Narasi Use Case "Setup Chatbot"

Ringkasan	Pemilik restoran memberikan otorisasi kepada sistem <i>chatbot</i> agar bisa membaca dan membalas pesan pada <i>Facebook Page</i> restaurannya
Aktor	Pemilik Restoran
Prasyarat	Berinteraksi dan menjawab permintaan pelanggan.
Deskripsi rangkaian utama	<ol style="list-style-type: none"> <li>1. Pemilik restoran mengeklik tombol otorisasi <i>chatbot</i> pada laman <i>Owner Dashboard</i>.</li> <li>2. Sistem menampilkan daftar <i>Facebook Page</i> yang dimiliki pemilik restoran.</li> <li>3. Pemilik restoran memilih salah satu <i>Facebook Page</i> yang ingin ditanamkan sistem <i>chatbot</i>.</li> <li>4. Pemilik restoran mengunjungi laman pengaturan <i>Facebook Page</i>, kemudian menambahkan tombol dengan kategori "Get in touch &gt; Send Message".</li> </ol>

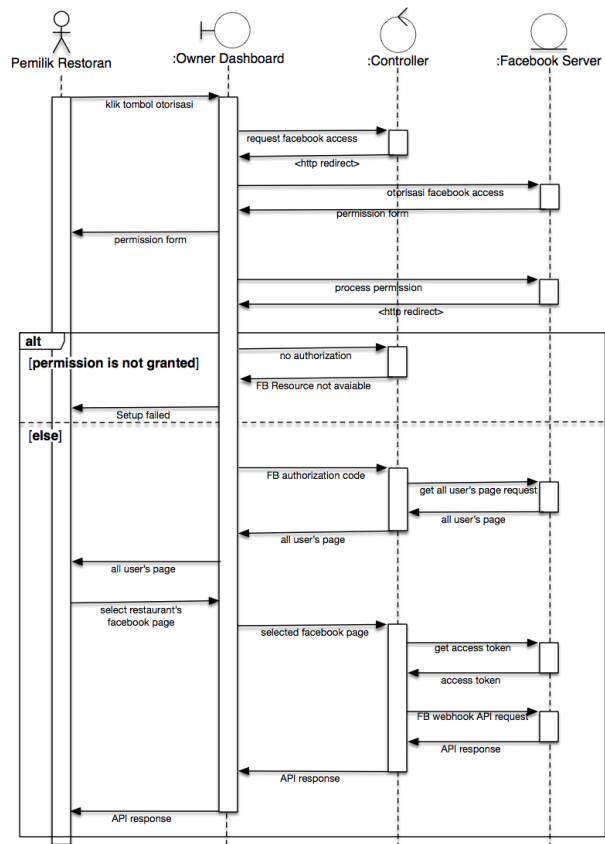
Narasi setiap dari *use case* yang pada Gambar 5 dibuat untuk memudahkan dalam fase-fase selanjutnya. Adapun salah satu contoh dari narasi *setup chatbot* adalah sebagaimana ditunjukkan oleh Tabel 2. Setelah melakukan pemodelan fungsional, tahap selanjutnya adalah melakukan pemodelan struktural, yaitu: dengan membuat *class diagram*. Model struktural berikut ini dibuat tanpa mengindikasikan bagaimana ia disimpan, dibuat, maupun dimanipulasi, sehingga difokuskan terhadap bisnisnya saja, tanpa dikacaukan dengan urusan teknis secara rinci.



Gambar 4. Class Diagram

Hasil analisis *class diagram* yang dilakukan menunjukkan bahwa terdapat lima relasi yang terdapat pada sistem usulan, yaitu kelas *Reservation*, *Reservation Assistance*, *Customer*, *Chatbot*, dan *Credit Card*.

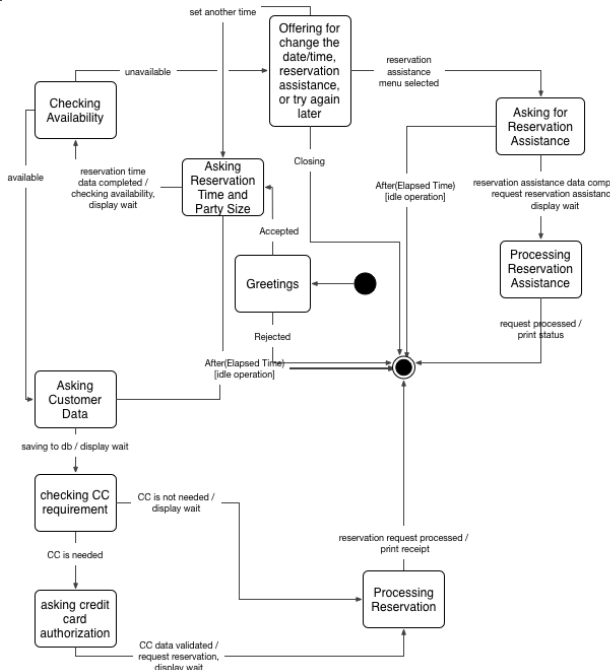
Tahap selanjutnya adalah melakukan pemodelan perilaku, yaitu dengan melakukan analisis atas *internal logic* pada sistem usulan tanpa menentukan bagaimana ia diimplementasikan secara fisik nantinya. Ada dua pemodelan perilaku yang dikerjakan, yaitu pemodelan interaksi antar objek dan pemodelan *dialog chatbot*. Pemodelan interaksi antar objek dalam sistem usulan terbagi dalam tiga proses, proses otorisasi *chatbot*, permintaan reservasi, dan permintaan bantuan reservasi. Adapun interaksi yang dilakukan antar objek pada tiga proses tersebut digambarkan dalam *sequence diagram* agar lebih mudah dipahami. Gambar 7 berikut ini adalah salah satu contoh dari analisis yang telah dilakukan.



Gambar 7. Sequence Diagram "Setup Chatbot"

Adapun pemodelan *dialog chatbot* dibuat dengan menggunakan *diagram statechart*. Dialog yang dilakukan oleh *chatbot* dengan pengguna ditentukan berdasarkan *states* atau keadaan-keadaan yang telah didefinisikan. Untuk setiap percakapan secara otomatis dimulai dengan *state* "greeting", kemudian *chatbot* memberikan penawaran apakah pengguna ingin melakukan reservasi, jika iya maka *state chatbot* berubah menjadi "asking reservation time" dan *chatbot* sekarang menanyakan kapan waktu reservasi yang diinginkan. Jawaban apa pun yang diberikan oleh pengguna akan dianggap sebagai keterangan waktu, karena saat ini *chatbot* sedang menanyakan kapan waktu reservasi yang diinginkan, dan akan berlanjutnya ke *state* selanjutnya jika keterangan waktu yang diberikan oleh

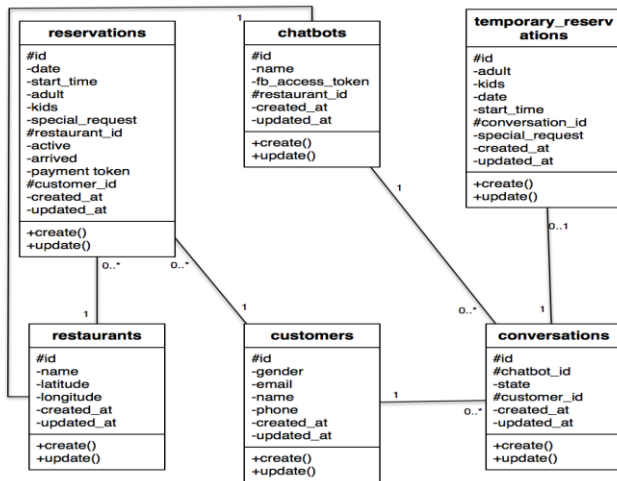
pengguna adalah *valid*. Secara garis besar, *states* yang terdapat *chatbot* digambarkan pada Gambar 8. Untuk *state* yang kompleks seperti “*Asking Reservation Time and Party Size*”, “*Asking Customer Data*”, dan “*Asking for Reservation Assistance*” digambarkan ke diagram terpisah agar lebih jelas lagi.



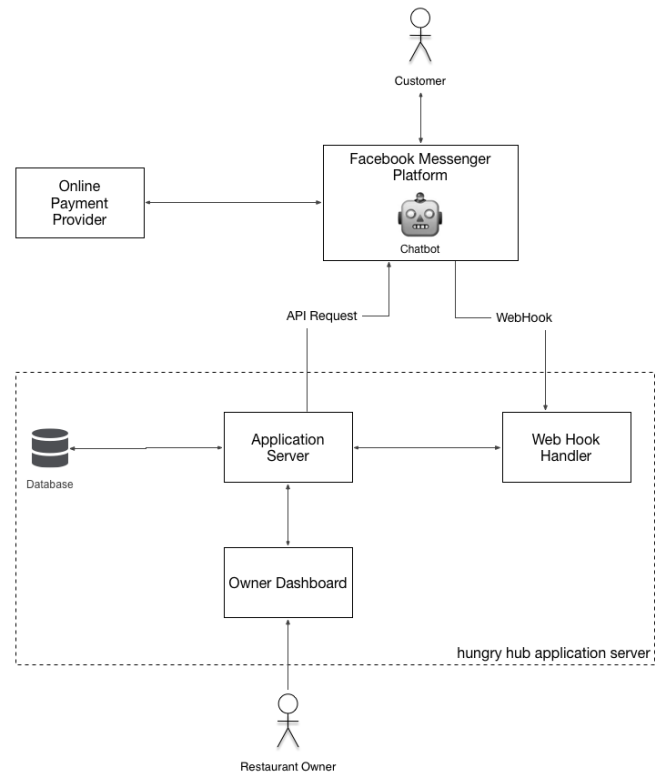
Gambar 8. Statechart Diagram Chatbot

C. Perancangan Sistem

Perancangan sistem dilakukan dengan merancang data, arsitektur, dan antarmuka. Tahapan yang dilakukan pada fase perancangan data adalah dengan mengubah *class diagram* yang telah dibuat pada fase pemodelan struktural ke dalam format *database* dengan melakukan pemetaan *class diagram* (Gambar 9) dan membuat *schema database*-nya. Secara umum, arsitektur sistem terlihat pada Gambar 10.



Gambar 9. Mapping Class Diagram



Gambar 10. Arsitektur Sistem

Ada dua jenis rancangan antarmuka yang dilakukan, yang pertama adalah rancangan antarmuka untuk sistem *Hungry Hub*, yaitu: dengan membuat rancangan perubahan antarmuka pada halaman *dashboard* pemilik restoran. Perubahan tersebut diperlukan untuk menambah tombol yang akan menampilkan sebuah *pop-up* otorisasi untuk mengakses *Facebook Page* yang Tabel 3.

Contoh Salah Satu Rancangan *Dialog Chatbot*

Context	Dialog	Hi, my name is <Bot Name>. Can I help you with table reservation at <Restaurant Name>? (Yes, Sure! ; No, Thanks)
Greetings	Accepted	Continue
	Rejected	No worries, Have a great day!
Asking date	Great, which date are you looking to reserve for? (Today ; Tomorrow ; I'll specify a date)	
	Today/Tomorrow	(Skip)
	Specify a date	Please enter the date you are interested in (e.g. 15 May)?
Validating date	Valid	Continue
	Invalid	I didn't get you, do you mind typing in the date again, we need it in the format of day followed by full month name (e.g. 18 August)?
Asking time	Please choose the preferred dining time? (18:00; 19:00 ; 20:00 ; Other)	
	18:00/19:00/20:00	(Skip)
	Other	OK, please tell me the time with 15 minutes interval format, e.g. 19:00, 19:15, 19:30, 19:45, 20:00, etc?

dimiliki oleh pemilik restoran. Adapun jenis rancangan yang kedua adalah rancangan *dialog* bagi *chatbot* (Tabel 3).

#### D. Implementasi Sistem

Pembuatan dan pengujian *chatbot* dilakukan dengan menggunakan model *finite state machine* agar bisa mengimplementasikan apa yang telah di rancang pada diagram *statechart*. Adapun pengujian *chatbot* dilakukan menggunakan metode *Test* dan *Behavior Driven Development* dengan tools *Cucumber*. Dokumentasi sistem juga dilakukan pada fase ini untuk memudahkan pengguna dalam menggunakan sistem yang baru dibuat. Contoh pengujian *Test Driven Development* menggunakan *tool* RSpec dalam pembuatan basis data ditunjukkan Gambar 11.

Chatbot
should validate that :access_token cannot be empty/falsy
should validate that :restaurant_id cannot be empty/falsy
should have many customers through conversations
should have many conversations
should belong to restaurant
should validate that :access_token is case-sensitively unique

table structure
contains `Restaurant` column with type integer
contains `Id` column with type integer
contains `Name` column with type string
contains `Created at` column with type datetime
contains `Fb page` column with type string
contains `Access token` column with type text
contains `Updated at` column with type datetime

Conversation
should validate that :customer cannot be empty/falsy
should validate that :chatbot cannot be empty/falsy
should belong to customer
should belong to chatbot

Gambar 11. Hasil Pengujian Basis Data

Sedangkan pembuatan dan pengujian *chatbot* dilakukan dengan menggunakan model *finite state machine* agar bisa implementasi sesuai dengan diagram *statechart*. Pengujian *chatbot* dilakukan menggunakan metode *Behavior Driven Development* dengan *tool* *Cucumber*. Contoh hasil pengujian ditunjukkan pada Gambar 12.

#### Feature: Modify Booking

Scenario: Customer creates valid reservation without CC and she never chat before but random		
Given a new Customer with facebook name "Akyu CAntik"		
And a new Bot with name "Mr. Hungry" that belongs to "Warung Tegal" restaurant		
But they never chat before		
And reservation doesn't require credit card		
And today is "6 June 2017"		
And customer contacts the bot on the first time		
And Inventory is available on "12 June" at "18:00" - "20:00" for 20 party size		
Then bot's state should be "greeting"		
And replied with:		
type	content	
text	Hi, my name is Mr. Hungry. Can I help you with table reservation at Wa	
qr button	Yes, Sure!	
qr button	No, Thanks!	
When customer selected "ACCEPT" qr button		
Then bot's state should be "asking date"		
And replied with:		
type	content	means
text	Great, which date are you looking to reserve for?	
qr button	Today	TODAY
qr button	Tomorrow	TOMORROW
qr button	I'll specify a date	CUSTOM
When customer replied with "12 Jun"		
Then bot's state should be "validating date"		
Then bot's state should be "asking time"		
And replied with:		
type	content	means
text	Please choose the preferred dining time?	

Gambar 12. Hasil Pengujian Chatbot

#### V. KESIMPULAN

Penelitian ini menghasilkan sistem *chatbot* pada sistem berjalan yang terintegrasi dengan platform *Facebook Messenger*. *Chatbot* yang dibuat mampu melayani permintaan reservasi oleh pengguna *Facebook* yang ingin melakukan reservasi pada situs *Hungry Hub*. *Chatbot* dibuat menggunakan metode *rule-based*, sehingga hanya mengerti maksud pesan yang diberikan oleh konsumen jika telah didefinisikan sebelumnya.

Untuk penelitian selanjutnya sebaiknya memberikan kemampuan bagi *chatbot* untuk mengerti pesan yang diberikan oleh konsumen dalam bahasa lainnya. Selain itu, sebaiknya *chatbot* juga mampu berinteraksi sebagai *Reminder* kepada konsumen untuk mencegah jikalau konsumen lupa. Metode lainnya seperti *neural network*, *machine learning*, atau semacamnya juga bisa digunakan agar kemampuan *chatbot* menjadi lebih baik lagi dalam menerjemahkan pesan yang diberikan oleh konsumen.

#### REFERENSI

- [1] R. P. Smith, "The Impact of Online Reservation Systems: For Chain Restaurants," University of Houston, 2011.
- [2] B. A. Shawar dan E. Atwell, "Accessing an Information System by Chatting," *Natural Language Processing and Information Systems*. Springer-Verlag, pp. 407-412, 2004.
- [3] E. Dohrmann, J. Hugi, R. Schneider, dan C. Eggenberg, "Bridging the virtual and the physical space: Kornelia—a chatbot for public libraries,"

- no. February 2017, pp. 0–7, 2010.
- [4] M. Lasek dan S. Jessa, “Chatbots for Customer Service on Hotels Websites,” *Information Systems in Management*, vol. 2, no. 2, hal. 146–158, 2013.
- [5] S. Sayed, R. Jain;, B. Lokhandwala;, F. Barodawala;, dan M. Rajkotwala;,, “Android based Chat-Bot,” *International Journal of Computer Applications*, vol. 137, no. 10, 2016.
- [6] E. Bahartyan;, N. Bahtiar, dan I. Waspada;, “Integrasi Chatbot Berbasis Aiml Pada Website E-Commerce Sebagai Virtual Assistant Dalam Pencarian dan Pemesanan Produk (Studi Kasus Toko Buku Online Edu4indo.Com),” *Jurnal Masyarakat Indonesia*, vol. 5, no. 10, 2014.
- [7] E. Nila S. C. P. dan I. Afrianto, “Rancang bangun aplikasi chatbot informasi objek wisata kota Bandung dengan pendekatan natural language processing (studi kasus di Dinas Kebudayaan dan Pariwisata Kota Bandung),” *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, vol. 4, no. 1, pp. 49-54, 2015.
- [8] B. A. Shawar dan E. Atwell, “Chatbots : Are they Really Useful?,” in *LDV-Forum*, 2007.
- [9] A. Dingli dan D. Scerri. Building a Hybrid: Chatterbot – Dialog System. In: Habernal I., Matoušek V. (eds) Text, Speech, and Dialogue. TSD 2013. Lecture Notes in Computer Science, vol 8082. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-642-40585-3\\_19](https://doi.org/10.1007/978-3-642-40585-3_19)
- [10] E. L. Amalia, D. W. Wibowo, “Rancang Bangun Chatbot Untuk Meningkatkan Performa Bisnis,” *Jurnal Ilmiah Teknologi Informasi Asia*, vol. 13, no. 2, pp. 137-142, 2019.
- [11] L. Zhang, “Building Facebook Messenger,” 2011. [Online]. Tersedia: pada:<https://www.facebook.com/notes/facebookengineering/building-facebook-messenger/10150259350998920>. [Diakses: 31-Mar-2017].
- [12] A. Joshi and M. J. Carman, “SarcasmBot: An open-source sarcasm-generation module for chatbots,” in *Fourth International Workshop on Issues of Sentiment Discovery and Opinion Mining (WISDOM)*, 2015, August 2015.