

# Before Order - A menu information providing chatbot service

1<sup>st</sup> Kang Minju  
dept. Information System  
Hanyang Univ.  
Seoul, South Korea  
kmj1997@gmail.com

2<sup>nd</sup> Lim Hyojin  
dept. Information System  
Hanyang Univ.  
Seoul, South Korea  
hyojin12288@gmail.com

3<sup>rd</sup> Choo Yedeun  
dept. Information System  
Hanyang Univ.  
Seoul, South Korea  
cyddddd1221@gmail.com

4<sup>th</sup> Christian Gärtner  
dept. Computer Science  
Hanyang Univ.  
Esslingen am Neckar, Germany  
christian.gaertner97@gmail.com

**Abstract**—Globalization attracted many newcomers to South Korea but our restaurant services have not kept up with the needs of foreign customers. Foreigners still have difficulties reading and understanding menus in traditional but also modern Korean restaurants. Therefore, we will develop ‘BeforeOrder’ by using the ‘KakaoTalk’ chatbot API and provide foreigners with a service to conveniently receive information about menus and dishes throughout their journey.

**Index Terms**—translator, chatbot

## I. INTRODUCTION

Many foreigners are often lost and embarrassed when eating out at restaurants as they can neither read the foreign menus nor can they understand what ingredients are used for specific dishes. Those things can make it hard to try and enjoy foreign food especially for people who have allergies and therefore have to ask the staff every time. But staff members, especially in more rural areas, often do not speak or understand English and cannot provide sufficient information. On the other hand, restaurants often simply do not have the space in menus to list the name of dishes and its ingredients in both, the foreign and the English language.

Whenever foreigners face this situation, they have to search and retrieve all the different information manually. However, foreigners who do not know ‘Hangeul’ will not be able to search directly on Google because they do not understand the dish names in a menu and often also do not have access to a ‘Hangeul’ keyboard. Even if someone searches for the information, it may not be properly categorized and recognized by a translator as they often do not distinguish dishes and it may happen that they simply translate the individual letters into English. Eventually the user may need to search more and more and ends up leaving the restaurant and eat someplace else.

We will try to solve this problem with ‘KakaoTalk’, a chat application that is essential in Korea. Since 2013, Kakao has provided an automated response API through KakaoTalk Plus Friends. Plus Friends can be added as a friend in KakaoTalk to receive various contents, benefits and information. Using this method, the user can conveniently install and delete the service provided by us and also use the functionality with an intensively tested and user-friendly interface.

A chatbot in general describes an artificial intelligence-based program, that analyzes the conversation with a user. Users of the messenger can naturally send messages or ask questions to a chatbot which in return provides a service that responds to the user and seems to communicate with him. There are multiple reasons as to why KakaoTalk chatbots are getting attention. First, it has a familiar UX design and developing a new UX can be very time-consuming and users may have difficulties adapting to a new design. Second, people tend to be reluctant to download new apps for a variety of reasons. Third, using a popular and well-known messenger improves accessibility. We think using a chatbot will increase the popularity of the service and make it more appealing to customers in our target audience.

*Before Order* target customers are foreigners that are not familiar with the Korean language and alphabet and therefore having problems eating out. The service is especially aimed at exchange students and employees who plan to stay in Korea in the long term instead of tourists that will mostly eat in international restaurants. *Before Order* follow this process:

Step 1: befriend *Before Order* as a Plus Friend in KakaoTalk

Step 2: provide information about the menu to the chat bot in form of images or text

Step 3: receive information about a dish by choosing from the menu list provided by *Before Order*

That way, customers can

- reach the service anytime
- receive information about dishes without searching multiple times
- do not have to register up for any service except befriend-ing *Before Order* on KakaoTalk

TABLE I  
USER ROLES

Role	Name	Task and description
User	Choo Yedeun	Assumes user point of view and guarantee a usable and user-friendly software
Customer	Lim Hyojin	Provides detailed software requirements and reviews delivered features
Software Developer	Christian Gärtner	Responsible for writing and developing software features and satisfying the needs of the customer
Development Manager	Kang Minju	Supervises the development of the service, managing of deadlines and evaluation of software features

## II. REQUIREMENTS

### A. Functional requirements

- A.1 The system will be usable on a smartphone – In order to enable the user to use the provided service while the user is moving around the system has to be reachable from a mobile platform. The user also shall be able to conveniently use a camera to take pictures of the menu without having to type and search for one dish name at a time.
- A.2 The system should be usable on notebooks, tablets and stationary computers – The user shall be able to use the service at home or similar environments to be able to gather information about restaurants and dishes and help with the selection of a place to eat.
- A.3 The system will run on Android, iOS and Windows 7/8/10 – The system shall run on the most popular operation systems in order to be accessible for as many users in the target audience as possible.
- A.4 The system will run within a common third-party software – The user shall be provided with an intensively tested and user-friendly interface. In addition, the user should not have to download a separate application to use the functionalities of this system but instead use this service as an extension to a familiar and popular application.
- A.5 The system shall provide contact information to enable the user to contact the support – The user shall be able to receive help in case of questions or problems with the service. In addition, the user should be able to report any bugs that he finds while using the service or give feedback about it.
- A.6 The user shall be able to save information about a dish – The user should be able to retrieve information about dishes that he once searched for without having to query the search again. The user should also be able to read the information after closing and reopening the application.

- A.7 The system shall support the Korean alphabet as input
- A.8 The system will display information about dishes in the English language and alphabet
- A.9 The user shall be able to select a dish in the analyzed menu and receive detailed information for this specific dish The system should provide information about the dishes in form of used ingredients, possible allergies and level of spiciness
- A.10 The system shall accept pictures and text as input – The user should be able to manually search for a dish if the analyzing of the picture returns no matches
- A.11 The system should distinguish between dish names and random letter clusters – The user may input pictures without a menu or the pictures that include random text in addition to the menu

### B. Working mechanisms

- 1) Register service  
User will search for *Before Order* in the KakaoTalk Plus Friend list and add the chatbot as a friend. Registration process is finished.
- 2) Input information  
Take a picture of a menu written in Korean alphabet and send it to the *Before Order* chat bot.
- 3) Recognition process  
The *Before Order* service analyzes the input picture or text and tries to recognize the dishes.
- 4) Select information  
If the user sent a whole menu he can select a single dish out of the list that *Before Order* provides and receive more detailed information for that specific dish.
- 5) Display detailed information  
*Before Order* provides the user with detailed information about the requested dish using public APIs.

## III. DEVELOPMENT ENVIRONMENT

### A. Choice of software development platform

TABLE II  
DEVELOPMENT TOOLS

Tool	Usage
JAVA	We've chosen Java as a overall basic language in developing <i>Before Order</i> because it is one of the general, object-oriented language and It is a language our team members are all familiar with.
Github	We decided to use Github because it provides the necessary management functions for software development including the basic functions of the Git such as bug tracking, functional requests, task management and etc. Also, we can easily share and develop program sources together with our team members.
Visual Studio	Visual studio is an integrated development environment(IDE) program made from Microsoft that includes many features as compiler, editor and debugging. We are going to utilize many various necessary functions of visual studio in practical implementation.
Slack	Slack is a collaborative tool that enables more efficient work and communication while developing software. We are going to take advantage of the work messenger functionality of slack and its convenient drag-and-drop format file sharing

#### B. Software in use

The similar software can be referred to as Samsung *Bixby vision*. *Bixby vision* automatically extracts text messages through camera lens and translate them into user-set languages in real time. In order to use *Bixby vision* menu analysis, we can just bring the camera lens to the menu. However there are two major differences that suggest why our *Before Order* is better than 'Bixby vision'. First, *Bixby vision* only provide literal interpretation of food name. Therefore people are more likely to have difficulties not knowing what exactly it means. However *Before Order* give more clear information as it aims to provide photo, detailed explanation and name of the food/menu in english. Lastly *Bixby vision* is limited to only samsung product but our *Before Order* is available on all platforms and devices since it will be implemented in chatbot.

#### C. Cost estimation

TABLE III  
DEVELOPMENT TOOLS

Tool	Cost
Azure Bot Service	Free
SQL database & server	16870.76 Won for 5GB / 10 DTU
Computer Vision API	2811.63 Won for 1000 transactions

The OS version in which we will develop the program is Window 10, and We will use our own laptop for developing our service.

We will also use cloud computing service using Azure cloud platform. The service we are using for making our service is:

- 1) SQL Database
- 2) Azure Bot Service
- 3) Computer Vision API : OCR api

We will connect our Azure Bot Service with Facebook messenger platform, and make senario in the Azure Bot Service. In order to store and manage the menu data, we will connect our Bot service with the SQL Database from Azure. We will make database for storing korean and english name of the menu and the details about menu (picture, ingredients, taste, spicyness, etc). Computer vision api will provide us texts from the picture that users send through the bot.

### IV. SPECIFICATIONS

#### A. Facebook Chatbot

We are using facebook chatbot. So this picture can show you communicate among user, server, database. The information provided to the user is stored in the database so that the server provides information when needed.

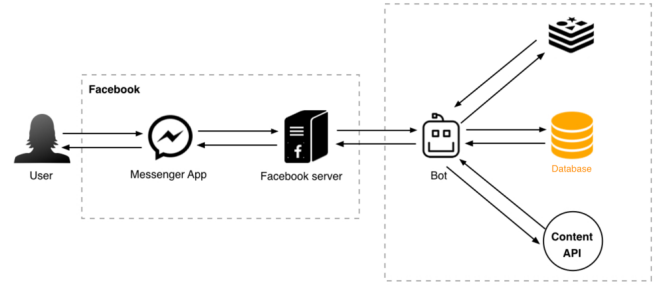


Fig. 1. Overview of the system

If you are using Facebook Messenger, you don't need a separate download. You can use the chatbot system by browsing the *Before Order* page on Facebook, finding our page and clicking the Sending Message button.



Fig. 2. Chatbot profil on Facebook

#### 1) Facebook messenger:

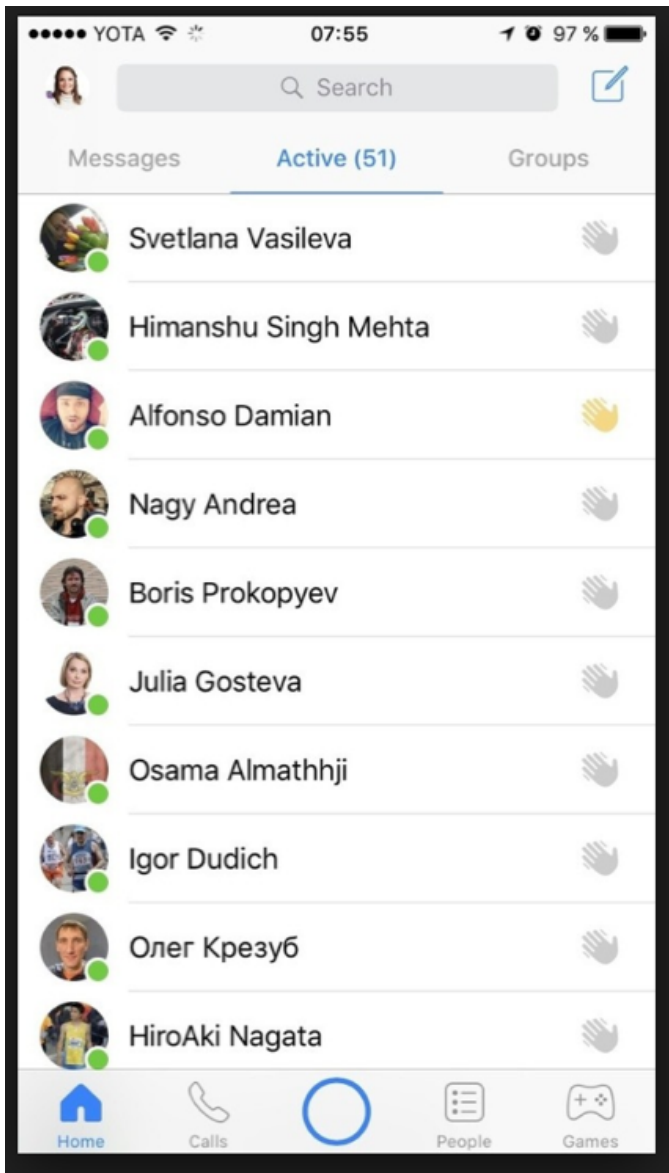


Fig. 3. Homescreen of the Facebook Messenger

If you select *Before Order* from the messenger list, we ask you what information you need to provide information.

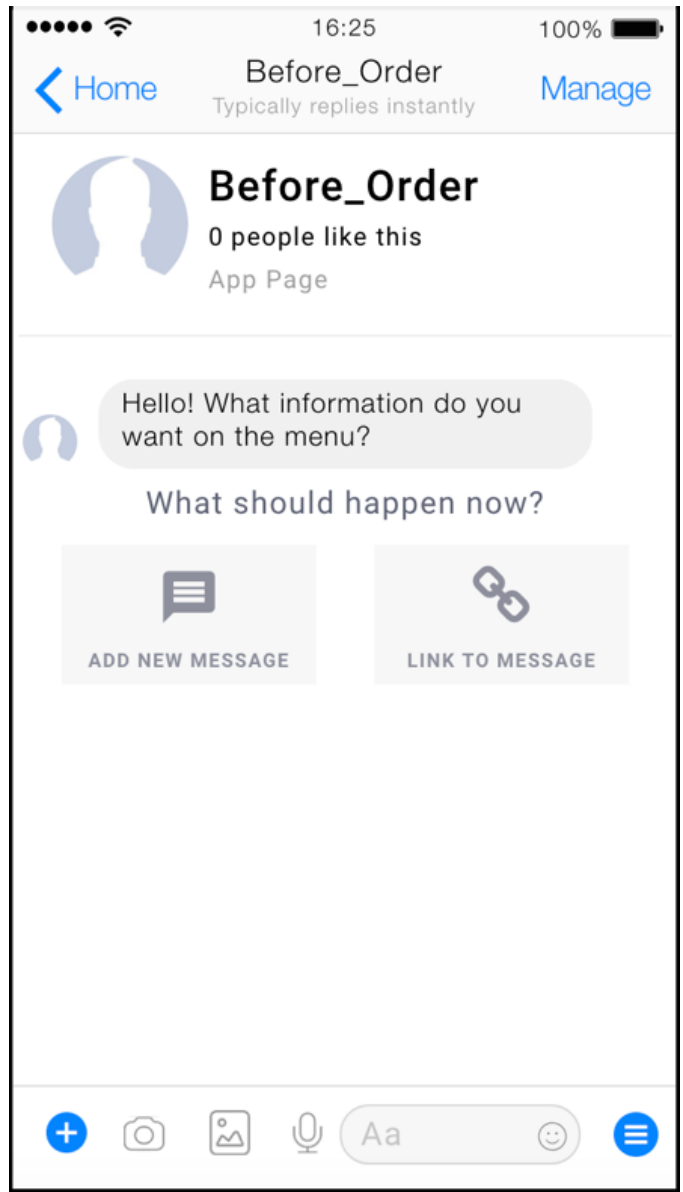


Fig. 4. Chatroom with the *Before Order* chatbot

2) *Chatbot input*: The user sends a picture with the dishes written in letters. Most menus include no pictures of the dishes but only the names.



Fig. 5. Message to the chatbot including a picture of a menu

We provide a list of those letters using the text recognition API in the image and allow the user select from the list.

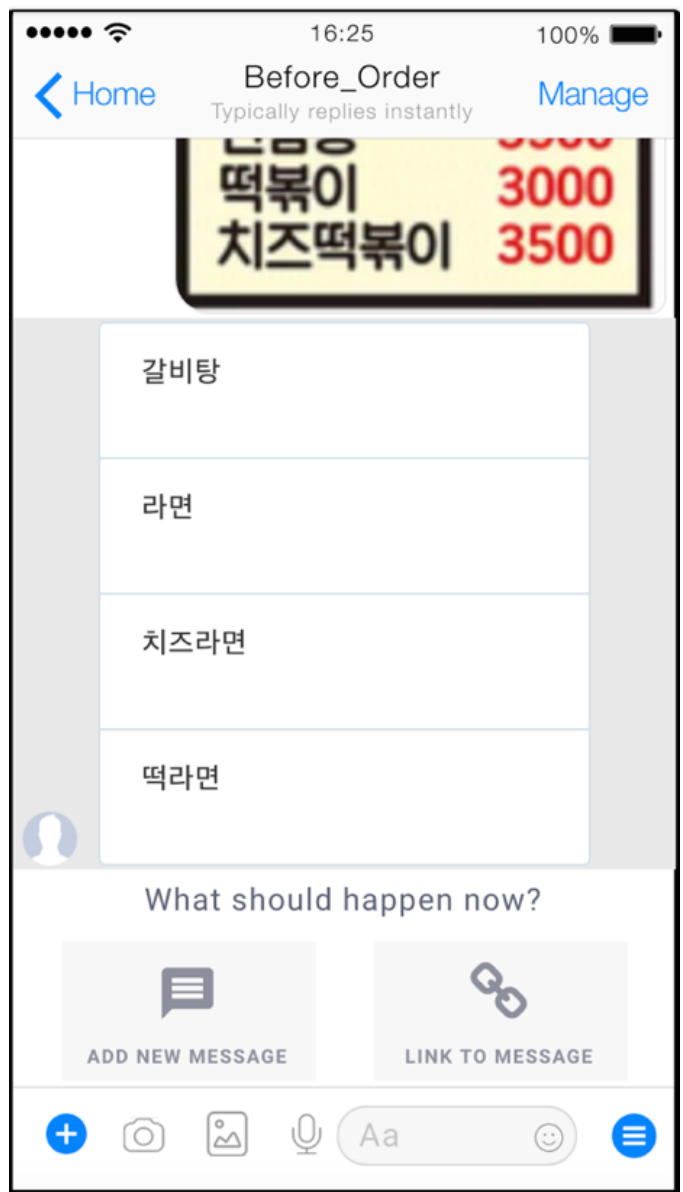


Fig. 6. Response of the chatbot with all the found dishes

The user selects the food for which information is to be provided in the list.

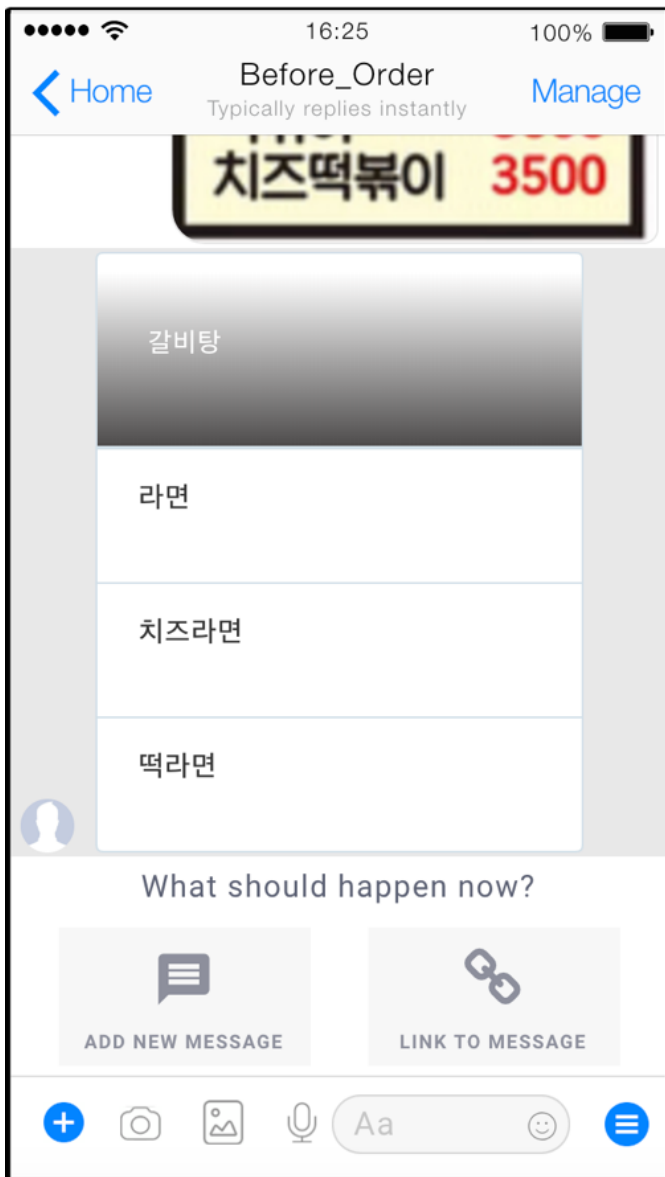


Fig. 7. User selecting a dish from the list

### 3) Processing of the input:

- Get the information from database
- Server that can run chat-bot
- Vision api connection

### 4) Chatbot output:

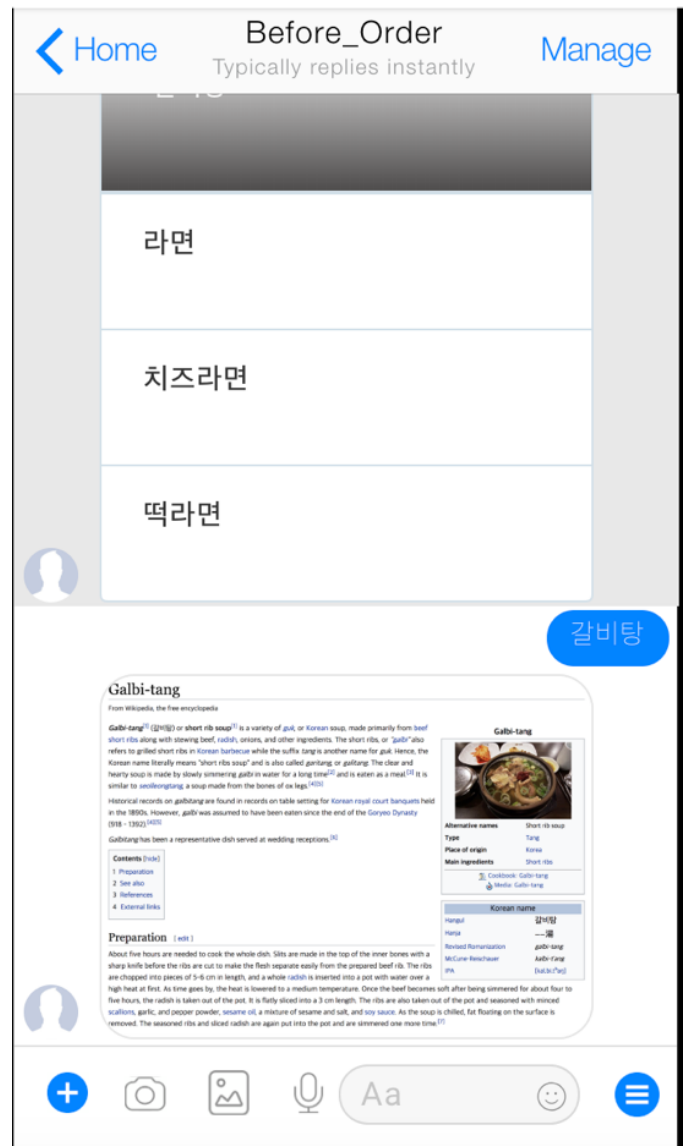


Fig. 8. Detailed informations about the selected dish

Although we can not represent it in this prototype, we plan to provide information from our internal database. Users can get photos, materials, and descriptions of food.

5) *Exit*: If you need information about other menus, please feel free to send us a message.