

# **Proposal for HI-Chat**

## **: A Chatbot for Hongik University Information Guidance**

지도교수	팀원	학번	C011147	이름	이승연
이장호			C111038		김지윤

### **1. Introduction**

Navigating university services, academic calendars, and campus facilities can be a daunting task for students, especially for freshmen and those returning from a leave of absence. Despite the availability of this information on the official Hongik University website, students often find it difficult to locate specific details quickly and efficiently. To address this issue, we propose the development of "HI-Chat," a chatbot system designed to provide convenient, real-time access to a variety of university-related information.

HI-Chat aims to improve the accessibility and user experience for students seeking information about university life, including campus facilities, academic schedules, administrative services, and welfare programs. This proposal outlines the system design, development plan, related work, and key references for the successful implementation of the HI-Chat project.

## **2. Motivation and Problem Statement**

Although Hongik University provides comprehensive information on its website, users often experience challenges such as poor navigation, scattered announcements, and delayed updates. This creates a gap in timely and effective communication, especially during course registration periods, scholarship applications, or when navigating administrative services.

By developing a chatbot that is accessible via the university website or popular messaging platforms such as KakaoTalk, we aim to simplify the information retrieval process. The chatbot will serve as an intuitive interface, reducing the time and effort needed to access important academic and campus-related information.

For example, if a user asks, “How do I apply for a national scholarship?” instead of just providing a general “scholarship-related website” link, the chatbot should guide the user directly to the official school website or application page for national scholarships.

Moreover, with the increasing reliance on mobile platforms, students demand quick, personalized, and reliable responses without having to dig through long webpages. A chatbot, operating 24/7, can help address this demand and enhance the overall student experience.

By doing so, users can quickly find the necessary information, making the service more efficient and useful.

## **3. Related Work**

Several universities have implemented chatbot systems to assist students. For example, Georgia State University's "Pounce" chatbot helped incoming students with enrollment, leading to a significant reduction in summer melt. Similarly, the University of Murcia developed a chatbot using IBM Watson to answer frequently asked questions.

If we look universities around us, Ewha Womans University has developed “Chat-E,” Seoul National University has built “SNU-bot,” and Chung-Ang University provides its own chatbot service called “CHARLI,” all of which are designed with their specific campus services in mind.

In terms of technology, many chatbots are built on decision trees or scenario-based logic, especially when the domain is limited and the user intents are relatively predictable. Tools like Dialogflow or Rasa enable developers to design effective question-response flows without requiring complex machine learning models.

These studies and tools demonstrate the feasibility and effectiveness of using rule-based or scenario-driven chatbots in educational settings. They also highlight the increasing importance of adopting conversational agents as a key part of student support infrastructure.

#### **4. System Overview**

HI-Chat will be a web- and messenger-integrated chatbot platform. The system will consist of the following components:

- 1) User Interface (UI): A web interface and KakaoTalk integration for ease of access.
- 2) Backend Server: Java-based backend (e.g., using Spring Boot) to handle logic, authentication, and routing.
- 3) Database: MySQL for structured storage of academic and administrative data.
- 4) Chatbot Engine: Scenario-based engine built using rule-based flows or frameworks such as Dialogflow.

The system architecture is modular to support future enhancements such as multilingual capabilities, additional APIs, or extended service integration.

## **5. System Features and Modules**

### 5.1. Campus Facility Guidance

- Search for classroom locations
- Guide to using the library
- Information on convenience facilities (e.g., cafeterias, lounges)

### 5.2. Academic Schedule and Course Registration

- View semester schedules
- Announcements for course registration
- Graduation requirements and mandatory course information

### 5.3. Administrative and Student Services

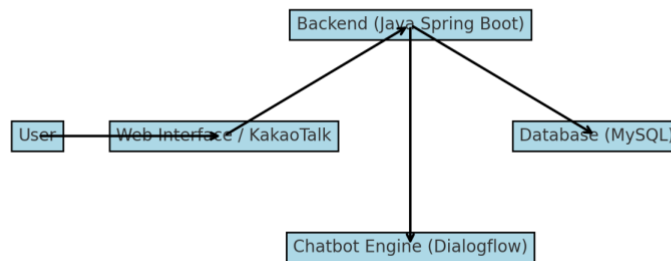
- Information on scholarships and student loans
- Guide to welfare and support programs
- Answers to frequently asked administrative procedures (e.g., leave of absence, tuition refund)

### 5.4. FAQ Automation

- Automated responses to frequently asked questions
- Constant updates through web scraping from official sources
- Support for both keyword-based and button-based quick replies

## 6. Technical Implementation

The development process consists of several key stages:



### 6.1. Requirement Analysis & Data Collection:

- Collect data from the university's official website and public announcements.
- Define chatbot conversation scenarios based on common student queries.
- Categorize questions into thematic areas such as academics, facilities, and support services.

### 6.2. Web Service Development:

- Set up a Java-based backend server using Spring Boot.
- Develop chatbot UI using HTML, CSS, and JavaScript.
- Construct a MySQL database to store queryable information.

### 6.3. Scenario-based Chatbot Development:

- Implement rule-based conversation flows using tools like Dialogflow or custom logic.
- Define intents and responses for common queries.
- Build fallback and escalation logic to handle unrecognized inputs.

#### 6.4. Testing and Refinement:

- Conduct usability testing with sample users.
- Refine scenarios and improve clarity of responses based on feedback.
- Optimize UI responsiveness and database performance.

#### 6.5. Deployment:

- Host the chatbot on the web and connect it to KakaoTalk via open API.
- Ensure scalability and minimal downtime through proper deployment strategies.

### **7. Team Roles and Responsibilities**

To efficiently manage the project, team members will take on specialized roles:

#### ***- Frontend Developer: 이승연***

- Design and implement the web interface for chatbot interaction
- Ensure responsive and user-friendly UI using HTML, CSS, and JavaScript
- Integrate KakaoTalk or other messaging platforms for chatbot deployment

#### ***- Backend Developer: 김지윤***

- Build and maintain the backend server using Java (Spring Boot)
- Design and manage the MySQL database
- Implement the logic for chatbot intent handling and scenario-based response flow

This clear division of responsibilities ensures parallel progress and efficient communication throughout the project.

## **8. Expected Outcomes and Contributions**

The HI-Chat system is expected to provide the following benefits:

- Improve the accessibility of information for Hongik University students.
- Reduce administrative workload by automating repetitive inquiries.
- Offer a scalable and adaptable system that can be maintained and expanded by future student developers.

Through this project, we aim to bridge the communication gap between the university and students using a practical, scenario-driven chatbot solution.

## **9. References**

1. IBM Watson Assistant Documentation. IBM Cloud Docs.
2. Georgia State University's Pounce Chatbot Case Study. EdSurge.
3. Dialogflow Documentation. Google Cloud Docs.
4. Rasa Open Source Documentation.
5. Jurafsky, D., & Martin, J. H. (2023). "Speech and Language Processing" (3rd ed.).
6. Chung-Ang University CHARLI Project Report (2022).
7. Seoul National University SNU-bot Project Blog.
8. Ewha Women University Chat-E Project Blog.