

# Ski Bot

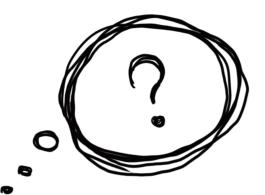
Kaicheng Jia

### Goals

#### Create a chatbot who can...

- Recommend ski resorts
- Provide weather and operating information





Already open?

Closed date?

Ticket prices?

Snowfall?

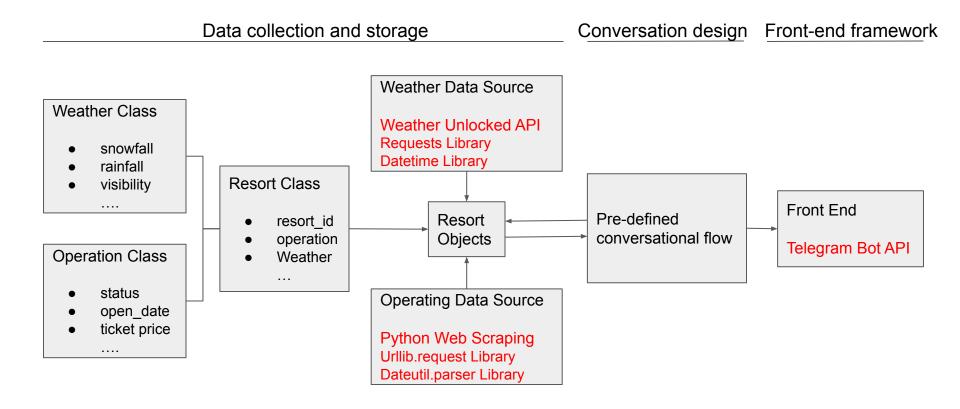
. . . . .

## **Problem Decomposition**

1. Real-time data collection and storage

Flow-based conversation design Front-end framework Can you tell me the weather in heavenly on 12/16/2021 5:12 PM V Okay, here's what I've found: Date: 12/16/2021 Weather: Patchy snow possible Snowfall (inch): 0.0 Real-time data Rainfall (inch): 0.0 Visibility (inch): 6.0 Humidity Level (miles): 30.08 Temperature (F): 33.0 Wind Speed: (mhp): 12.0 5:12 PM Do you have any other questions? 5:12 PM Conversation design Front-end Write a message...

## Tools & Techniques



#### **Future Extension**

- Data Caching: Query each time vs Query at bootstrap + periodic update
- Data Storage: Database vs In-memory
- Conversation Design: Pre-defined vs NLP
- **Testing:** Manual **vs** Unittest

#### References

- Github link: <a href="https://github.com/Kaicheng1995/SkiBot">https://github.com/Kaicheng1995/SkiBot</a>
- Weather Unlocked API: <a href="https://developer.weatherunlocked.com/documentation/skiresort">https://developer.weatherunlocked.com/documentation/skiresort</a>
- Telegram Bot API: <a href="https://core.telegram.org/bots/api">https://core.telegram.org/bots/api</a>
- Url for web scraping: <a href="https://www.skiresort.info/">https://www.skiresort.info/</a>
- Datetime Library: <a href="https://docs.python.org/3/library/datetime.html">https://docs.python.org/3/library/datetime.html</a>
- Tutorial for Python web scraping: <a href="https://realpython.com/python-web-scraping-practical-introduction/">https://realpython.com/python-web-scraping-practical-introduction/</a>
- Tutorial for Python API: <a href="https://www.dataquest.io/blog/python-api-tutorial/">https://www.dataquest.io/blog/python-api-tutorial/</a>