



# DICQUIZ

SSAFY 김성찬  
SSAFY 이민우

# INDEX

1

챗봇 소개

2

개발과정

3

실행화면

4

개선점



# 1. 챗봇 소개





# 기본 정보

---

- Chatbot Name : DICQUIZ ( DICTIONARY + QUIZ )
  - 어학 능력 향상을 위한 퀴즈 챗봇
  - 궁금한 단어도 찾고, 문제도 풀고!



## 2. 개발 과정





# 개발 배경

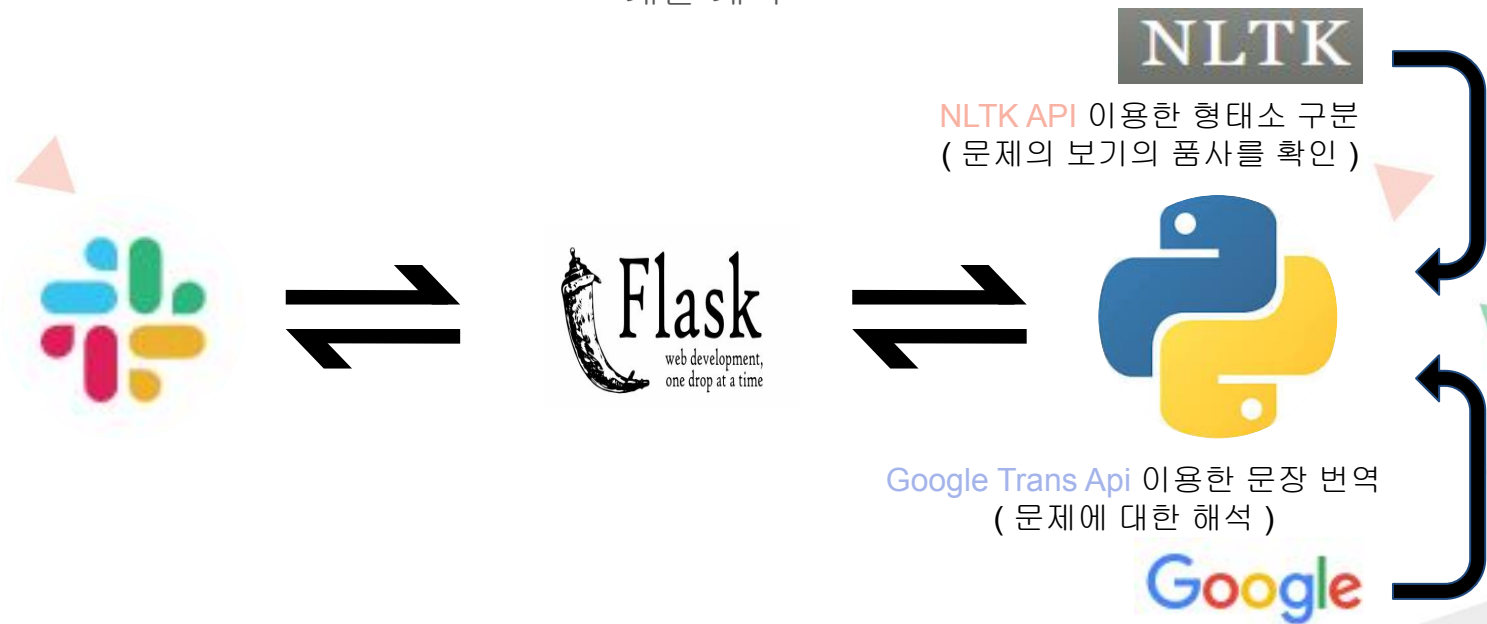
---

- 다양한 아이디어를 생각해봤지만,
  - 처음에는 추천 봇을 생각했다
  - 다른 주제의 챗봇은 없을까?
  - 자가 발전을 위한 챗봇은 어떨까?



# 개발 과정

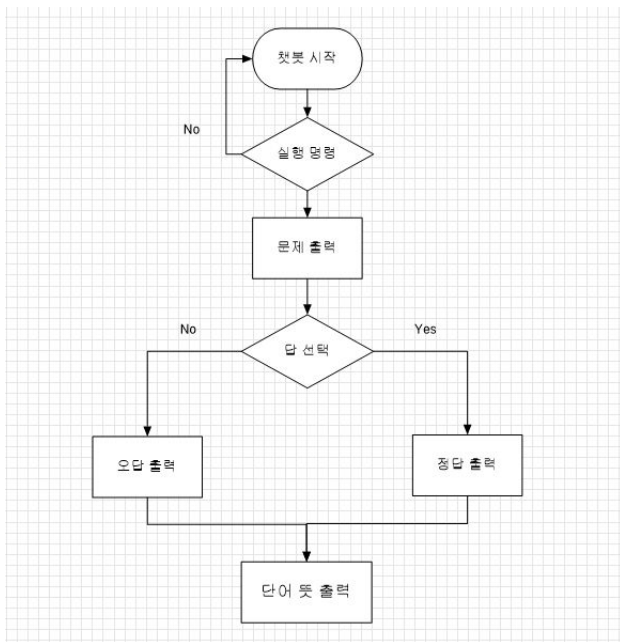
개발 계획





# 개발 과정

## 개발 계획



- **Search\_??**  
: 단어 검색 기능
- **endict**  
: 문제 출력
- **Show\_success**  
: 맞춘 단어 목록 (상위 10개)
- **Show\_fail**  
: 틀린 단어 목록





# 개발 과정

---

데이터

tire ★

1. 피곤하게 하다 2. 타이어 3. 지치다 4. 싫증나다

**Ddum 사전**

동사

명사



# 개발 과정

---

개발 중 어려웠던 사항

단어를 어떻게 가져올 것인가?

문제에 대한 다른 답을 어떤 것으로 가져올 것인가?

크롤링 할 때, 단어마다 링크가 달라 어떻게 리다이렉트 할 것인가?

# 작업화면 Pycharm

```
44 def create_json(dictionary, filename):
45     with open(filename, 'w') as file:
46         json_string = json.dumps(dictionary)
47         file.write(json_string)
48
49 def create_dict(filename):
50     with open(filename) as file:
51         json_string = file.read()
52         return json.loads(json_string)
53
54 def make_tuple_sorted(filename):
55     createdict = create_dict(filename)
56     make_tuple = []
57     for value in createdict:
58         make_tuple.append((value, createdict[value]))
59     return sorted(make_tuple, key=itemgetter(1), reverse=True)
60
61 def result_update(filename):
62     global target
63     createdict = create_dict(filename)
64     if target in createdict:
65         createdict[target] += 1
66     else:
67         createdict[target] = 1
68     create_json(createdict, filename)
```

## 파일 관련 메소드

```
70 def show_stats(text):
71     if text in "showsucces":
72         tuplesort = make_tuple_sorted("success.json")
73     else:
74         tuplesort = make_tuple_sorted("fail.json")
75     result_dict = {}
76
77     for value in tuplesort:
78         if len(result_dict) > 9:
79             break
80         result_dict[value[0]] = value[1]
81
82     top_list = []
83     for value in result_dict:
84         top_list.append("{}: {}회".format(value, result_dict[value]))
85
86     return "\n".join(top_list)
87
88 def create_word():
89     global url, source_code, soup
90     randomvalue = randint(100000, 200000)
91     wordid = "000" + str(randomvalue)
92
93     url = "https://dic.daum.net/word/view.do?wordid=ekw" + wordid
94     source_code = urllib.request.urlopen(url).read()
95     soup = BeautifulSoup(source_code, "html.parser")
96
97     word = ""
98     for data in soup.find("span", class_="txt_cleanword"):
99         word = data.strip().lower()
100     return word
```

```
243 def search_word(text):
244     try:
245         message = _crawl(text[20:].replace(" ", "%20"))
246     except IndexError:
247         return "찾을 수 없는 단어입니다."
248     result1 = []
249     result1.append(text[20:] + "의 의미는 [" + message + "] 입니다.")
250     result1 = result1 + mean
251     return "\n".join(result1)
```

사전 데이터를 랜덤으로  
불러오기

단어 검색 메소드

# 작업화면 Pycharm

```
102 def word_interpret(word):
103     translator = Translator()
104     return translator.translate(word, src='en', dest='ko').text
105
106 def length_check(word):
107     if len(word) > 130:
108         return True
109     else:
110         return False
111
112 def include_check(word, target):
113     if word in target:
114         return True
115     else:
116         return False
117
118 def change_morpheme(word):
119     target = nltk.word_tokenize(word)
120     temp = []
121     for value in target:
122         temp[0] = temp[0] + value + " "
123
124     res = nltk.pos_tag(temp)
125     return res[0][1]
```

```
if include_check(word, quiz):
    total.append("문제 : " + quiz.replace(word, "_____"))
else:
    continue
total.append("해석 : " + word_interpret(quiz))
quest_list.append(word)
target_morpheme = change_morpheme(word)
target = word
break
```

API 사용

( 형태소 구분, 한/영 번역 )

결과 출력

링크 리다이렉트

문제 출력

```
127 def output_result(isSuccess, message):
128     global target, mean
129     if isSuccess:
130         result_update("success.json")
131         result = ["*정답입니다!*"]
132     else:
133         result_update("fail.json")
134         result = ["*틀렸습니다!*"]
135         result.append("정답은 " + target + "입니다")
136         result.append(target + "의 의미는 [" + message + "] 입니다.")
137         result = result + "<" + mean + "|" + "자세히..." + ">"
138         target = ""
139     return result
140
141 def create_soup_with_url(url):
142     handler = urllib.request.urlopen(url)
143     source_code = handler.read().decode('utf-8')
144     return BeautifulSoup(source_code, 'html.parser')
145
146 def create_soup_with_query(query):
147     url = 'https://dic.daum.net/search.do?q=' + query
148     return create_soup_with_url(url)
149
150 def create_soup_with_word_id(query):
151     global mean
152     mean = []
153     for data in query.find_all("strong", class_="tit_cleansch"):
154         mean.append("https://dic.daum.net/word/view.do?wordid=" + data.get('data-tlars-id'))
155     return create_soup_with_url(mean[0])
```



### 3. 실행화면



# 실행화면 Slack

실행 시, **endict** 입력!  
잘못 입력 시, 안내문구 출력!

@dictbot sear



dictbot APP 12:46 PM

문제풀기 : 'endict'  
단어찾기 : 'search\_??'  
정답top10보기 : 'show\_success'  
오답top10보기 : 'show\_fail'



김성찬 8:26 AM  
@dictbot endict



dictbot APP 8:26 AM

문제 : a study on the effective emergency care using an automated external \_\_\_\_\_(aed) in the pre-hospital stage  
해석 : 병원 전 단계에서 자동화 된 외부 제세동 (aed)을 이용한 효과적인 응급 처치에 관한 연구  
1번: defibrillation  
2번: overcareful  
3번: sporting  
4번: windburn  
'1번', '2번', '3번', '4번' 중에서 골라주세요.

문제를 맞추면,  
정답임을 알리고  
맞춘 단어에 대한 뜻과 링크 출력

문제를 틀리면,  
정답을 알려주고  
틀린 단어에 대한 뜻과 링크 출력

옳은 시작어를 입력할 시,

크롤링한 데이터에서  
랜덤으로  
문제를 화면에 출력



김성찬 8:26 AM  
@dictbot 1번



dictbot APP 8:26 AM

정답입니다!  
defibrillation의 의미는 [1. 세동 제거 2. 제세동] 입니다.  
<https://dic.daum.net/word/view.do?wordid=ekw000199986>



김성찬 8:26 AM  
@dictbot 4번




dictbot APP 8:26 AM


틀렸습니다!  
정답은 \*marsh fern\*입니다  
marsh fern의 의미는 [처녀고사리] 입니다.  
<https://dic.daum.net/word/view.do?wordid=ekw000102838>


# 실행화면 Slack

크롤링 할 수 있는 단어가  
입력되면  
의미와 사전 링크를 출력

실행 시, **endict** 입력!  
잘못 입력 시, 안내문구 출력!

 김성찬 2:38 AM  
@dictbot en

 **dictbot** APP 2:38 AM  
문제풀기 : 'endict'  
'정답top10보기' : 'showsuccess'  
'오답top10보기' : 'showfail'

 김성찬 12:46 PM  
@dictbot search\_Asdadfqq

 **dictbot** APP 12:46 PM  
찾을 수 없는 단어입니다.

크롤링 할 수 없는 단어가  
입력되면  
예외처리



김성찬 12:46 PM

@dictbot search\_success



**dictbot** APP 12:46 PM

success의 의미는 [1. 성공 2. 성과 3. 달성 4. 성공한 사람 5. 대히트] 입니다.

<https://dic.daum.net/word/view.do?wordid=ekw000162533>



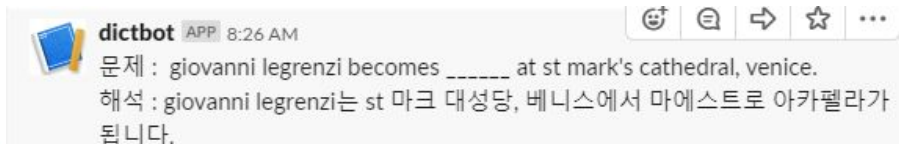
# 개선점





# 개선해야 할 사항

## ➤ 번역의 부족함



예문을 받았을 때, 해석이 매끄럽지 않은 경우가 발생

## ➤ 코드의 가독성을 위한 소스 정리

## ➤ 단어장 추가



# Q n A

[https://github.com/Lucehn/ssafy2\\_bootcamp](https://github.com/Lucehn/ssafy2_bootcamp)



**Thank you**