

- 대화의 시작은 면접관이 질문을 하는것으로 시작한다. 내가 응답을 하면 면접관은 해당 응답에 대한
- . 내가 응답을 하면 면접관은 해당 응답에 대한 평가를 제시하고, 하단에 [꼬리질문/모범답변/다음질문] 버튼을 생성한다. . 꼬리질문 클릭시
- 3. 꼬리질문 클릭시
 a. 질문, 답변을 기반으로 한 꼬리질문을 생성하여 질의한다.
 4. 모범답변 클릭시
 - a. 질문, 답변을 기반으로 사용자의 답변을 평가한다. . 다음질문 클릭시
- 5. 다음질문 클릭시 a. 새로운 질문을 생성한다.
- 꼬리질문, 모범답변, 다음질문 버튼 노출시 메시지입력 및 보내기는 disabled

취준 에이전트	
~답변평가~	
면접 꼬리질문 모범답변 다음	질문
~꼬리질문~	
면접 관 ~답변~	
	사용 자
메시지입력	보내기

질문 + 사용자 답변을 기반으로 꼬리질문을 생성한다.

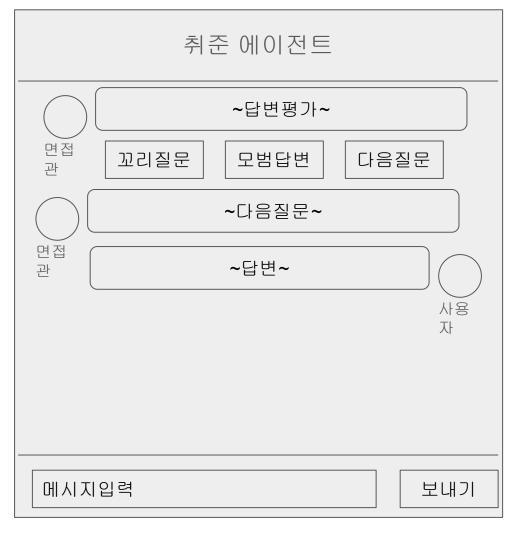
2. 사용자 답변이 입력되면 그에 대한 답변평가 [꼬리질문, 모범답변, 다음질문] 버튼을 동일하게 노출하다

1. 사용자가 꼬리질문을 클릭하면 면접관은 지난

. 사용사 납변() 꼬리질문, 모든 노출한다.

취준 에이전트	
	~답변평가~
_{면접} _관 고리질문	모범답변 다음질문
	~모범답변~
면접 관 고리질문	모범답변 다음질문
메시지입력	보내기

- 사용자가 모범답변을 클릭하면 면접관은 지난 질문 + 사용자 답변을 기반으로 모범답변을 생성하고, [꼬리질문, 모범답변, 다음질문] 버튼을 노출한다.
 꼬리질무 모범답변 다음질무 버튼 노출시
- 2. 꼬리질문, 모범답변, 다음질문 버튼 노출시 메시지입력 및 보내기는 disabled



사용자가 다음질문을 클릭하면 첫번째 flow와 같다.

```
1안
Enums
                contentType: 'question' | 'answer' | 'modelAnser'
                speakerType: 'agent' | 'user'
DTO
               ChatHistoryDTO
                                id : string;
                                children : List<ChatHistoryDTO> | null;
                                type : contentType;
                                speaker: speakerType;
                                content: string;
                AnswerRequestDTO
                                questionId: string;
                                content : string;
                QuestionDTO
                                id: string;
                                content : string;
                AnswerDTO
                                id : string;
                                content : string;
api 명세
               (GET) /chatHistory
                              payload : null
                               response : List<ChatHistoryDTO>
                (POST) /answer
                              payload : AnswerRequestDTO
                     b.
                               response : null
               (GET) /followUp
                               payload : {
                                questionId: string;
                               response : QuestionDTO
                     b.
                (GET) /question
                     a.
                              payload : null
                     b.
                              response : QuestionDTO
               (GET) /modelAnswer
                               payload : {
                                questionId: string;
                               response : AnswerDTO
```

```
2안
Enums
               contentType: 'question' | 'answer' | 'modelAnser'
               speakerType: 'agent' | 'user'
DTO
               ChatHistoryDTO
                               id : string;
                               type : contentType;
                               speaker: speakerType;
                               content: string;
               AnswerRequestDTO
                               questionId: string;
                               content : string;
               QuestionDTO
                               id: string;
                               content : string;
               AnswerDTO
                               id : string;
                               content : string;
api 명세
               (GET) /chatHistory
                              payload : null
                     b.
                              response : List<ChatHistoryDTO>
      2.
                (POST) /answer
                              payload : AnswerRequestDTO
                     a.
                    b.
                              response : null
     3.
               (GET) /followUp
                              payload : {
                               questionId: string;
                    b.
                              response : QuestionDTO
      4.
               (GET) /question
                              payload : null
                     a.
                              response : QuestionDTO
                    b.
      5.
               (GET) /modelAnswer
                              payload : {
                               auestionId: strina:
                              response : AnswerDTO
```