**Poll System – Poll service:**

**Models:**

**Answer:**

* id: Optional[int] = None
* question\_id: int
* user\_id: int
* choice\_id: int

**AnswerRequest:**

* title: str
* choice:str

**AnswerResponse:**

* title: str
* choice:str

**Choice:**

* id: Optional[int] = None
* question\_id: int
* choice:str

**Question:**

* id: Optional[int] = None
* title: str

**QuestionRequest:**

* title: str
* first\_choice: str
* second\_choice: str
* third\_choice: str
* forth\_choice: str

**QuestionResponse:**

* id: Optional[int] = None
* title: str
* choices: List[str]

**ChoiceCountByQuestionResponse:**

* title: str
* choices: List[ChoiceCountResponse]

**ChoiceCountResponse:**

* choice:str
* choice\_count:int

**UserAnswerResponse:**

* title: str
* choice:str

**End Points:**

**path=”question”**

* get\_survey() -> List[QuestionResponse] | get(path=”/”)
* get\_by\_id(question\_id) -> QuestionResponse | get(path=”/{question\_id}”)
* create(question\_request) -> QuestionResponse | post(path=”/”)
* update(question\_id, question\_request) -> QuestionResponse |put(path=”/{question\_id}”)
* delete(question\_id) | delete(path=”/{question\_id}”)

**path=”answer”**

* submit\_survey(user\_id, List[ answer\_request]) -> List[str] |
* create(path=”/{user\_id}”)
* update(user\_id, answer\_request) -> str | put(path=”/{user\_id}”)
* delete\_by\_user\_id(user\_id) | delete(path=”/{user\_id}”)

**path=”stats”**

* get\_choice\_count\_by\_question\_id(question\_id) -> ChoiceCountByQuestionResponse | get(path=”/choice\_count\_by\_question”)
* get\_user\_count\_by\_question\_id(question\_id) -> str | get(path=”/user\_count\_by\_question”)
* get\_user\_answers\_by\_user\_id(user \_id) -> UserAnswerResponse | get(path=”/user\_answers”)
* get\_user\_answer\_count\_by\_user\_id(user\_id) -> str | get(path=”/user\_answer\_count”)
* get\_all\_questions\_and\_choices\_selected\_count() -> List[ChoiceCountByQuestionResponse] |get(path=”/all\_question\_and\_choices\_selected\_count”)

**Service:**

**Question\_service:**

* get\_all() -> List[Question]
* get\_survey() -> List[QuestionResponse]
* get\_by\_id(question\_id) -> Question
* get\_by\_title(title) -> Question
* get\_question\_and\_choices\_by\_question\_id(question\_id) -> QuestionResponse
* create(question\_request) -> QuestionResponse
* update(question\_id, question\_request) -> QuestionResponse
* delete(question\_id)

**Choice\_service:**

* get\_by\_question\_id(question\_id) -> List[Choice]
* create(choice)
* delete(choice\_id)

**Answer\_service:**

* submit\_survey(user\_id, answers) -> List[str]
* create(answer)
* update(user\_id, answer\_request) -> str
* process\_answer(user\_id, answer\_request) -> str
* delete\_by\_question\_id(question\_id)
* delete\_by\_user\_id(user\_id)

**Stats\_service:**

* get\_choices\_selected\_count\_by\_question\_id(question\_id) -> ChoiceCountResponse
* get\_user\_count\_by\_question(question\_id) -> str
* get\_user\_answers\_by\_user\_id(user\_id) -> List[AnswerResponse]
* get\_answer\_count\_by\_user\_id(user\_id) -> str
* get\_all\_questions\_and\_choices\_selected\_count() -> List[ChoiceCountByQuestionResponse
* get\_choices\_selected\_count\_by\_question(question) -> ChoiceCountByQuestionResponse

**Repository:**

**Question\_repository:**

* get\_all() -> List[Question]
* get\_by\_id(question\_id) -> Optional[Question]
* get\_by\_title(title) -> Optional[Question]
* create(question: Question) -> Question
* update(question: Question) –> Question
* delete(question\_id)

**Choice\_repository:**

* get\_by\_question\_id(question\_id: int) ->List[Choice]
* create(choice)
* update(choice)
* delete(choice\_id)

**Answer\_reposititory:**

* get\_by\_question\_id\_and\_user\_id(question\_id, user\_id) -> Optional[Answer]
* create(answer)
* update(answer)
* delete\_by\_question\_id(question\_id)
* delete\_by\_user\_id(user\_id)

**Stats\_repository:**

* get\_choice\_selected\_count(choice\_id) -> int
* get\_user\_count\_by\_question\_id(question\_id) -> int
* get\_user\_answers\_by\_user\_id(user\_id) -> List[AnswerResponse]
* get\_answer\_count\_by\_user\_id(user\_id) -> int