Developer manual

In this document we will talk about an application that adding question, these questions are of one of three types slider, smiley and star, in addition to adding questions, you can edit and delete an existing question, then the questions are collected for **survey.**

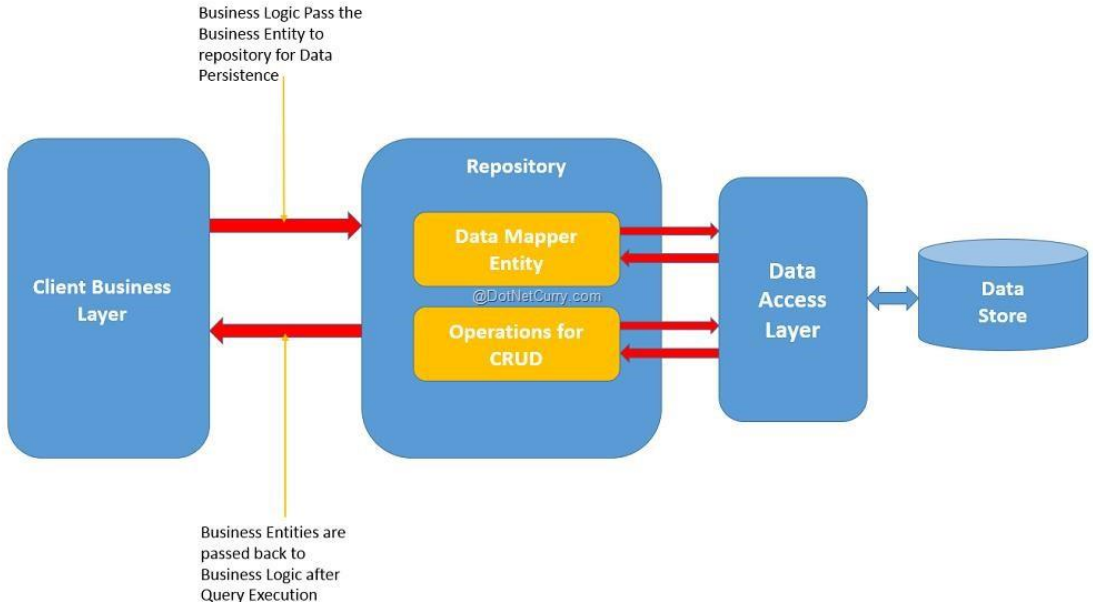
Survey questions can use either a closed-ended or open-ended format to collect answers from individuals. And you can use them to gather feedback from a host of different audiences, including your customers, colleagues, prospects, friends, and family.

Note: A closed-ended question includes a predefined list of answer options, while an open-ended question asks the respondent to provide an answer in their own words.

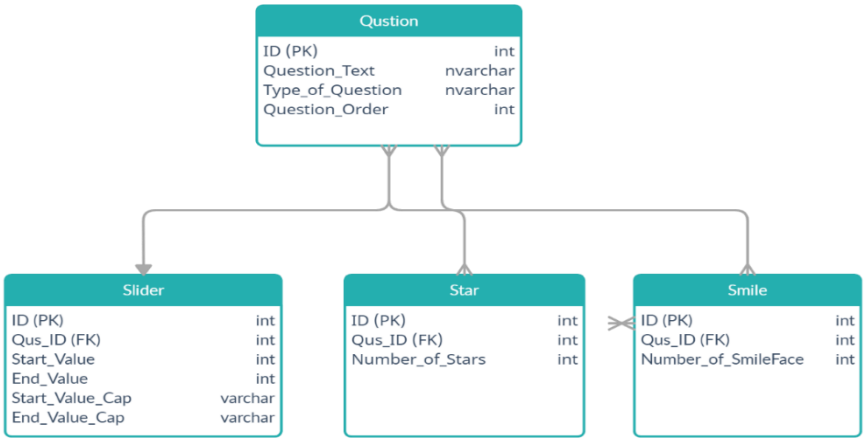
This application consists of a database to save questions and a windows form to view and edit the questions, this application is divided into layers and works on **repository design pattern.**

Repository design pattern

we can say that a **repository design pattern** acts as a middleman or middle layer between the rest of the application and the data access logic. That means a **repository pattern** isolates all the data access code from the rest of the application.



Database:



1- Database contain four tables

Table Question

- The table question is a base table it contains records of all types of questions it contains the common properties between all questions, every table wants to be question will derive table question by using question ID as a foreign key.

Table question consist of four columns:

1. **ID**
   - Primary key of table
2. **Question_Text**
   - It is the text of question
3. **Type_Of_Question**
   - It is the answer type of question

4.Question_Order

- It is the order of question between all questions

2- Table Slider

- The table slider is derived from table question will takes the properties from question table in addition to slider properties, every slider records have Qus_ID as foreign key from table question, each slider becomes a questions but not all question becomes a slider, the relation between table question and table slider is one to one.

Table slider consist of sex columns:

1.ID

- Primary key of table

2.Qus_ID

- Foreign key from table question

3. Start value

- The lowest value of the slider range (by default the slider will have a minimum of 1)

4.End value

- The highest value of slider range (by default the slider will have a maximum of 100)

5.Start caption

- It is the word symbolizing the answer in the start value range.

6.End caption

- It is the word symbolizing the answer in the end value range.

3- Table Smile

- The smile table is derived from question table will takes the properties from question table in addition to smile properties, every smile records have Qus_ID as foreign key from table question, each smile becomes a questions but not all question becomes a smile, the relation between table question and table smile is one to one.

Table smile consist of three columns:

1.ID

- Primary key of table

2.Qus_ID

- Foreign key from table question

3.Number_Of_Smile

- It is to answer the question through faces, the number of smile face from 2 to 5.

4-Table Stars

- The table star is derived from question table will takes the properties from question table in addition to star properties, every star records have Qus_ID is foreign key from table question, each star becomes a questions but not all question becomes a star, the relation between table question and table star is one to one.

Table star consist of three columns:

1.ID

- Primary key of table

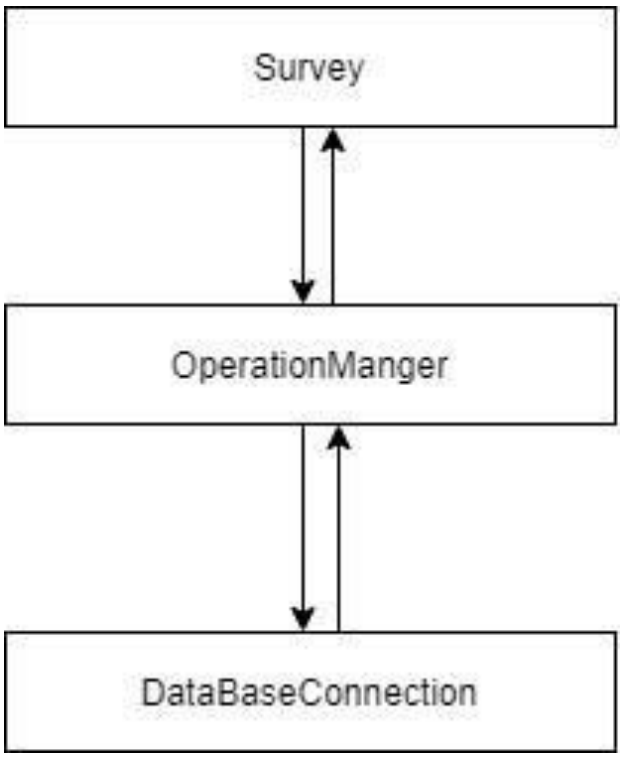
2.Qus_ID

- Foreign key from table question

3.Number_Of_Smile

- It is to answer the question through stars, and it is from 1 to 10.

Layers for application:



Database layer:

The database layer contains serval functions will access on database:

1.AddQuestion:

- **AddQuestion**
- **AddSlider**
- **AddSmile**
- **AddStar**

1.1.AddSlider(Question NewQuestion)

- This function take a question type slider will add it in database, first will call function AddQuestion() will add the question in table question and return succeeded if added and error code if an error occur in database, then check the result of add question is succeeded will add slider question in slider table and return to manger layer succeeded if added and error code if an error occur in database

1.2.AddSmile(Question NewQuestion)

- This function take a question type smile will add it in database, first will call function AddQuestion() will add the question in table question and return succeeded and error code if an error occur in database, then check the result of add question if succeeded will add smile question in smile table and return to manger succeeded and error code if an error occur in database

1.3.AddStart(Question NewQuestion)

- This function take a question type star will add it in database, first will call function AddQuestion() will add the question in table question and return succeeded and error code if an error occur in database, then check the result of add question if succeeded will add star question in star table and return to manger layer succeeded and error code if an error occur in database.

2.Edit Question:

- **EditSlider**
- **EditSmile**
- **EditStar**
- **EditQuestion**

2.1.EditSlider(Question QuestionWillEdit)

- This function take a question will edit as parameter, first will call function EditQuestion to edit question in table question in database and return succeeded and error code if an error occurs in database while edit in table question, then will check if return succeeded edit a question in table slider and return to manger layer succeeded and error code if an error occurs in database while edit in table slider.

2.2.EditSmile(Question QuestionWillEdit)

- This function take a question will edit as parameter, first will call function EditQuestion to edit question in table question in database and return succeeded error code if an error occurs in database while edit in table question, then will check if return succeeded edit a question in table smile and return to manger layer succeeded and error code if an error occurs in database while edit in table smile.

2.3.EditStar(Question QuestionWillEdit)

- This function take a question will edit as parameter, first will call function EditQuestion to edit question in table question in database and return succeeded and error code if an error occurs in database while edit in table question, then will check if return succeeded edit a question in table star and return to manger layer succeeded and error code if an error occurs in database while edit in table star.

3.Delete Question:

- **DeleteSlider**
- **DeleteSmile**
- **DeleteStar**
- **DeleteQuestion**

3.1.DeleteSlider(Question QuestionWillDelete)

- This function take a question will delete as parameter, first will delete the slider from table slider, then call delete the question from table question and return succeeded and error code if an error occurs in database while delete in table question then return the result return succeeded and error code if an error occurs in database.

3.2.DeleteSmile(Question QuestionWillDelete)

- This function take a question will delete as parameter, first will delete the smile from table smile, then call delete the question from table question and return succeeded when delete question and error code if an error occurs in database while delete in table question then return the result return succeeded if delete smile and error code if an error occurs in database.

3.3.DeleteStar(Question QuestionWillDelete)

- This function take a question will delete as parameter, first will delete the star from table star, then call delete the question from table question and return succeeded if delete question and error code if an error occurs in database while delete in table question then return the result return succeeded and error code if an error occurs in database.

4. GetQuestionFromDataBase(ref List)

- This function take a list as reference, will select the question from all tables then add it as objects of question in list the function will return succeeded when getquestion and error code if an error occurs in database while getting the question from tables.

2. Operation Manger layer:

This layer manager between database layer and UI layer, this layer takes the questions from UI for validation then take this question to database layer, when database layer finish will return the result if succeeded or fail will return to manger then manger return the result to UI then UI show message for user if succeeded or fail.

The manger layer contains serval functions will access on database layer:

1. **RefreshData**
2. **AddQuestion**
3. **EditQuestion**
4. **DeleteQuestion**
5. **GetQuestion**

2.1.RefreshData()

- This function for refresh list every 10 minutes if list is changed and call this function from UI, first will create background thread, then call function checkForRefresh() this function check if my list is changed or not if changed will refresh the data automatically, when the application is close thread stop.

2.2. AddQuestion (Question NewQuestion)

- This function for take the question from UI as parameter, then check the question which type of question after check will send the question to function add in database this function return result of adding to UI if added successfully return succeeded and return error code if not added.

2.3. EditQuestion (Question QuestionWillEdit)

- This function for take the question from UI as parameter, then check the question which type of question after check will send the question to function edit in database this function return result of editing to UI if edited successfully return succeeded and return error code if not edited.

2.4. DeleteQuestion (Question QuestionWillDelete)

- This function for take the question from UI as parameter, then check the question which type of question after check will send the question to function delete in database this function return result of deleting to manger then manger send the result to UI if deleted successfully return succeeded and return error code if not deleted.

2.5.GetQuestion (ref List)

- This function takes list from manger by reference to add all the questions from database layer then send the list to UI to show it, if getting the question successfully return succeeded and return error code if not get from database.

3. User interface layer:

This layer is for interacting between UI and user, contains some buttons to add, edit, delete, change language and display screen, this layer takes question from operation manager then here also check the validation of each variables for each question, this layer sends the question to add or delete or edit to manger and show the user if the operation completed successfully or not by message box.

4. BaseLog dl

This dll for write any errors for any layers in logfile using function Log(string message) this message when error occur you send exception message to Log function to write the message in logfile, the logfile have a default name is **Log.txt** and his location **bin->debug**.

5. Question dll:

This dll that contain data for application, four classes:

1. Abstract class Question.
2. Class Slider.
3. Class Smile.
4. Class Star.

The class slider, smile and star inherited from class Question.

Appendix database error code:

Name error code	Error code
Succeeded	0
ErrorInDataBase	500
ErrorConnectionString	501
ErrorInAddQuestion	502
ErrorInSelectionQuestion	503
ErrorInEditQuestion	504
ErrorInDeleteQuestion	505
ErrorInGetQuestion	506
ErrorInOperation	507

Appendix manger error code:

Name error code	Error code
Succeeded	0
ErrorInManger	400
ErrorInMangerAdd	401
ErrorInMangerEdit	402
ErrorInMangerDelete	403
ErrorInMangerGetQuestion	404