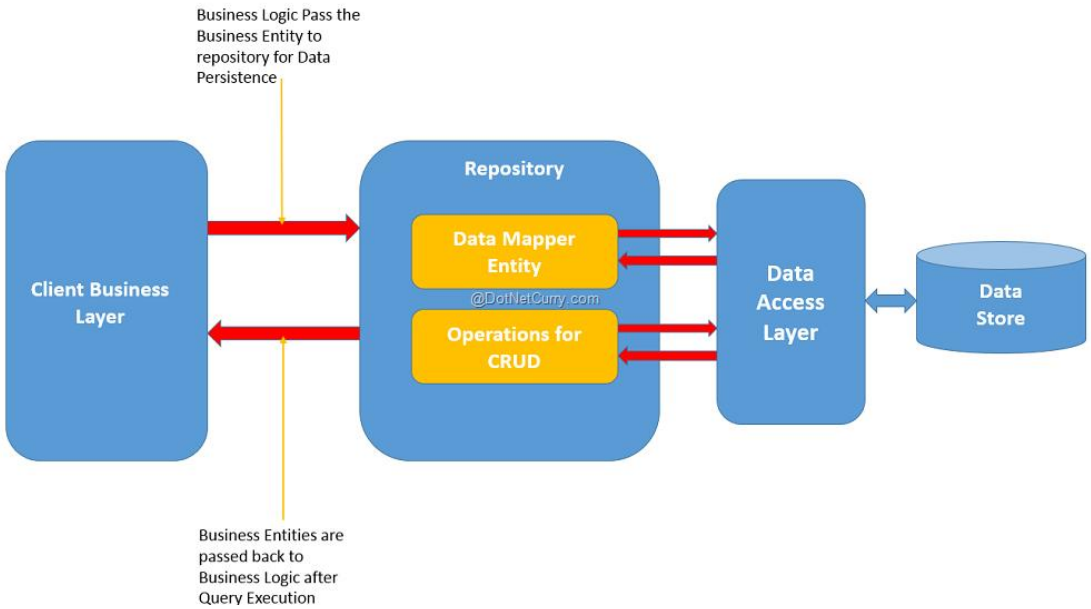
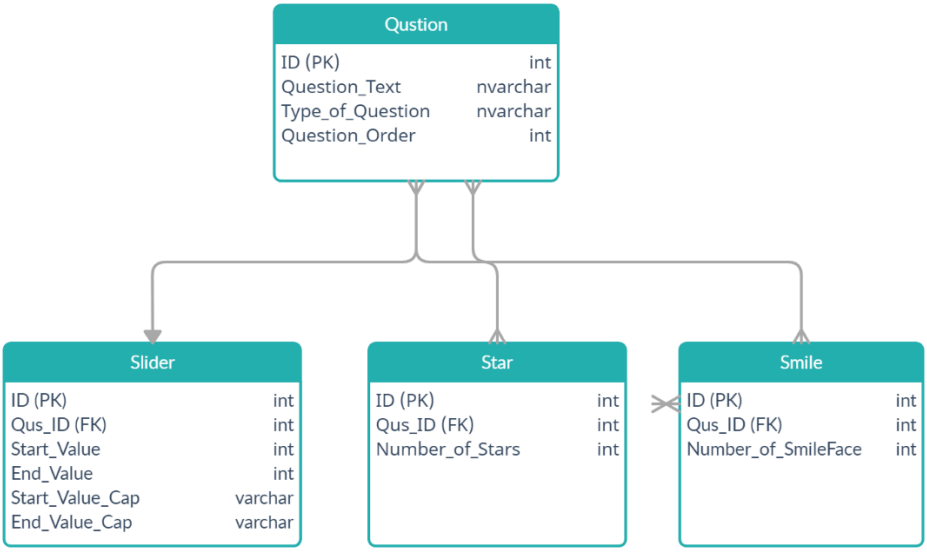
Developer manual

- This application consists of a database to save data and a windows form to view and edit the data, 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:



Database contain four tables

1- Table Question

- This table contain ID is primary key and it gives tables this primary key as foreign key to access the main table, which is the question, when the take this primary key, they will have the behavior of the question table, it contains three elements that take these elements in each question:

1. **Question_Text**
    - It is the text of question.

2. **Type_Of_Question**
    - It is the answer type of question.
3. **Question_Order**
    - It is the order of the question between the questions.

2- Table Slider

- This table contain ID is primary key and contain Qus_Id is foreign key from main table question, that’s mean every slider is question, it contains four elements each slider have this four elements in addition to the element of the question,
  Elements for slider:
    1. **Start value**
        - The lowest value of the slider range (by default the slider will have a minimum of 1)
    2. **End value**
        - The highest value of slider range (by default the slider will have a maximum of 100)

3.Start caption

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

4.End caption

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

3- Table Smile

- This table contain ID is primary key and contain Qus_Id is foreign key from main table question, that’s mean every smile is question, it contains one element each smile has element in addition to the element of the question,
  Element of smile:
  1. **Number of smile faces**
     - It is to answer the question through faces, and it is from 2 to 5.

4-Table Stars

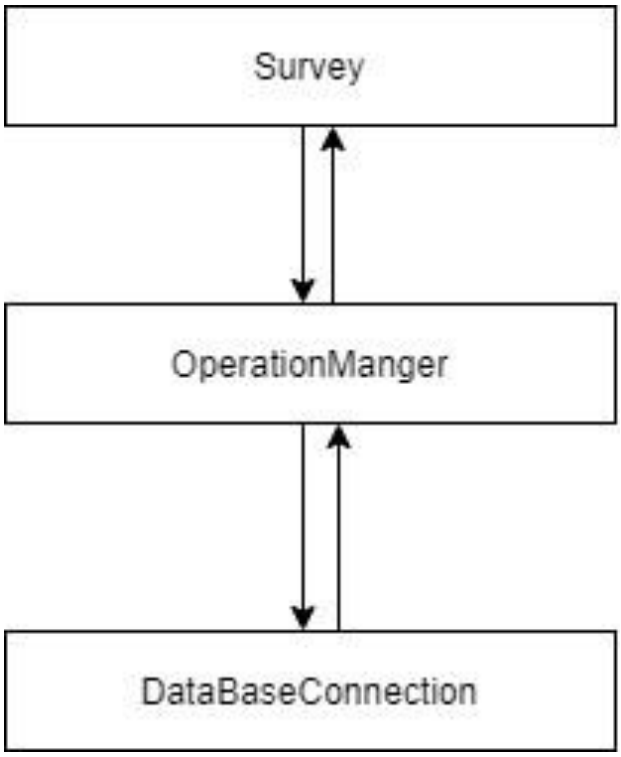
- This table contain ID is primary key and contain Qus_Id is foreign key from main table question, that’s mean every star is question, it contains one element each star has element in addition to the element of the question,

Element of star:

1.Number of stars

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

Layers for application:



1- Database layer:

The database layer contains several functions:

Add Question:

1.AddSlider
2.AddSmile
3.AddStar
4.AddQuestion

This functions for add question in any type in database, every type of question has function for add and all types call AddQuestion then call their own function return number for succeed or fail to operation manger delete question.

Edit Question:

1.EditSlider
2.EditSmile
3.EditStar
4.EditQuestion

This functions for edit question in any type from database every type of question has function for edit and all types call EditQuestion then call their own function and return number for succeed or fail to operation manger edit question.

Delete Question:

1.DeleteSlider
2.DeleteSmile
3.DeleteStar
4.DeleteQuestion

This functions for delete question in any type from database every type of question has function for delete and all types call DeleteQuestion then call their own function and return number for succeed or fail to operation manger delete question.

GetQuestionFromDataBase:

This function for get question from database to the list and return list to operation manger.

2. Operation Manger layer:

This section connects the database and (UI) layer get data from database and send data to UI and get data from UI and send the data to database and have thread for refresh data in period of time and resend new data to UI, and have several functions:

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

This functions will take the data from UI then the manger layer will send it to the database layer, and take data from database layer and send it to UI layer.

3. User interface layer:

This section is for interacting with the user and contains several buttons and a display screen, and this section takes data from operation manager `and` here also check the conditions of each variables for each question.

4. BaseLog dll:

This dll for print any errors for any layers in logfile using function Log() and you can change the location and the name of logfile using variable filename and his location variable filepath .

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.

All these classes inherited from class Question.