**Q2: Implementation of chapter 7: -**

Chapter 7 gives us detail information about Requirement Engineering, Non-functional requirements, Collaborative Requirements gathering, Use cases and about UML diagrams of Use case, Class Diagram, State Diagram, Analysis Patterns, Requirements Monitoring, validating Requirements. These concepts are used to understand the requirements provided by the stakeholder and implement them in the project, knowing the scope of the requirements and discussing them with the stakeholder to validate the requirement is one of the main learning in the chapter 7. The seventh exercise is a separate exercise unlike the exercises from one to six, it is not interlinked. In this exercise we must build a tic tac toe game with white background and mention the player names in the bottom, and once the game is completed the name of the player should be displayed as the winner and an option should be given to the user, that whether he wants to restart the game. In this we must investigate every viewpoint of the requirement i.e., use case of the chapter 7 and make sure that the requirements are been fulfilled which is to program the game in python. We should also keep in mind about the scope of the requirements which gives us the validation of the requirement.

Code Snippets:

1. In the below snippet we are initializing the board and designing the board according to the interest. The color of the symbols and thickness of the symbols are initiated here in the below part of the snippet

Text

Description automatically generated

1. In the snippet below we are arranging the X and O symbols, initializing the position of the X and O when clicked on the part of board

A screenshot of a computer

Description automatically generated with medium confidence

1. In the below part of the code the after-game page is designed i.e., who has won the game and an option is given the user that would the user wants to restart the game, if we clicks yes, then the game restarts and No it displays Thank you.

Graphical user interface, text, application, email

Description automatically generated

1. The below snippet is the logical part of the program, this decides who is the winner def is\_winner the function which checks the pattern and announces the winner and there is another function is the match is tie i.e., where there is no pattern recognized then it is tie. The once the game either the winner name is displayed or the tie is announced between the players

A picture containing text

Description automatically generated

1. The below code is if the user clicks yes as he’s preference when he is asked to restart the game. And calling the main function to execute the code.

Text

Description automatically generated

Output Screenshots: -

Table

Description automatically generated

A picture containing table

Description automatically generated

**Graphical user interface, text, application

Description automatically generated**

**YouTube Link of the execution video: -** [**https://youtu.be/BwwladAe-Gg**](https://youtu.be/BwwladAe-Gg)