SDP

MOBILE - APP DEVELOPMENT

SOURCE CODE

TOPIC : Tic-Tac-Toe GAME APP

TEAM MEMBERS :

ASHWINTH SINGH . P - R17CS055
AJAY . S - R17CS012
AMAL MIRWAIZ - R17CS018
ANNAY MAITY - R17CS035

About The App :

Tic-Tac-Toe is a classical paper-and-pencil game for 2 players. You were probably used to play Tic-Tac-Toe game during your childhood. The rules are quite simple : each player sets a X or O on a 3x3 grid. The player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row wins the game.

Creating the Board

First step is to create the Board for the Tic-Tac-Toe game. The *Board* class will store the elements of the grid in an array and will contain a boolean indicating if the game is ended or no.

The *play* method will let you to set the mark of the *currentPlayer* on the grid at a given (x,y) position. A *changePlayer* method will be used to change the current player for the next play. Besides, a *computer* method is defined to let the user to randomly place a mark on the grid. Finally, we define a *checkEnd* method to check if the game is ended. The game is ended if there is a winner or a draw : all the cases of the grids are filled and no one wins the game.

Creating the Board
First step is to create the Board for the Tic-Tac-Toe game. The Board class will store the elements of the grid in an array and will

contain a boolean indicating if the game is ended or no.

The play method will let you to set the mark of the currentPlayer on the grid at a given (x,y) position. A changePlayer method will be used to change the current player for the next play. Besides, a computer method is defined to let the user to randomly place a mark on the grid. Finally, we define a checkEnd method to check if the game is ended. The game is ended if there is a winner or a draw : all the cases of the grids are filled and no one wins the game.

Rendering the Board on the Screen

Next step is to create a *BoardView* class to render our Board on the screen. Our *BoardView* will extend the *View* class and we will draw the *Board* and its elements on the *Canvas* object associated. It is a good way to discover how to draw simple shapes on a *Canvas* of a specific *View* too.

Furthermore, we must manage the touch events of the users on the *Board* to let it to play to our Tic-Tac-Toe game. For that, we override the *onTouchEvent* method from the *View* parent class. In that method, we convert a point touched on the screen to a case on our grid. Then, we make the play on the *Board* object. After that, we need to call the gameEnded method of the parent activity if the game is ended to display the win dialog to the user. If not, we make the play for the computer. Like you can see, the heart of the logic game will be located in this method.

Creating the UI for our Game

The biggest part of the user interface of our Tic-Tac-Toe game is managed in the *BoardView* class. So, we just need to set our *BoardView* component into a *RelativeLayout* parent *View* in our layout file :

Starting a new Game

To star a new game, the user will have to click on a load item in the action bar of our application. So, we add the item in a main.xml menu file under /res/menu :

Assemble all the pieces of the puzzle

Last step is to assemble all the components created previously in the *MainActivity* class. In the *onCreate* method, we create the *Board* object and then we pass it in parameter of the *BoardView* got from the main layout of the application. Then, we connect the new game item of the action bar with the *newGame* method of the *Board* object to create a new game when the user will click on it. Finally, we define the *gameEnded* method which was called in the *BoardView* object.

Playing to Tic-Tac-Toe Game

Now, we can try our game and play to the famous Tic-Tac-Toe game . The game works great and finally, we win the game against the computer which is logical because our Artificial Intelligence (AI) is really basic .

FOR THE CODE , THE FILE IS ATTACHED TO THE MAIL

THANK YOU ,

TEAM :

Ashwinth Singh . P - R17CS055

Ajay . S - R17CS012

Amal Mirwaiz - R17CS018

Annay Maity - R17CS035