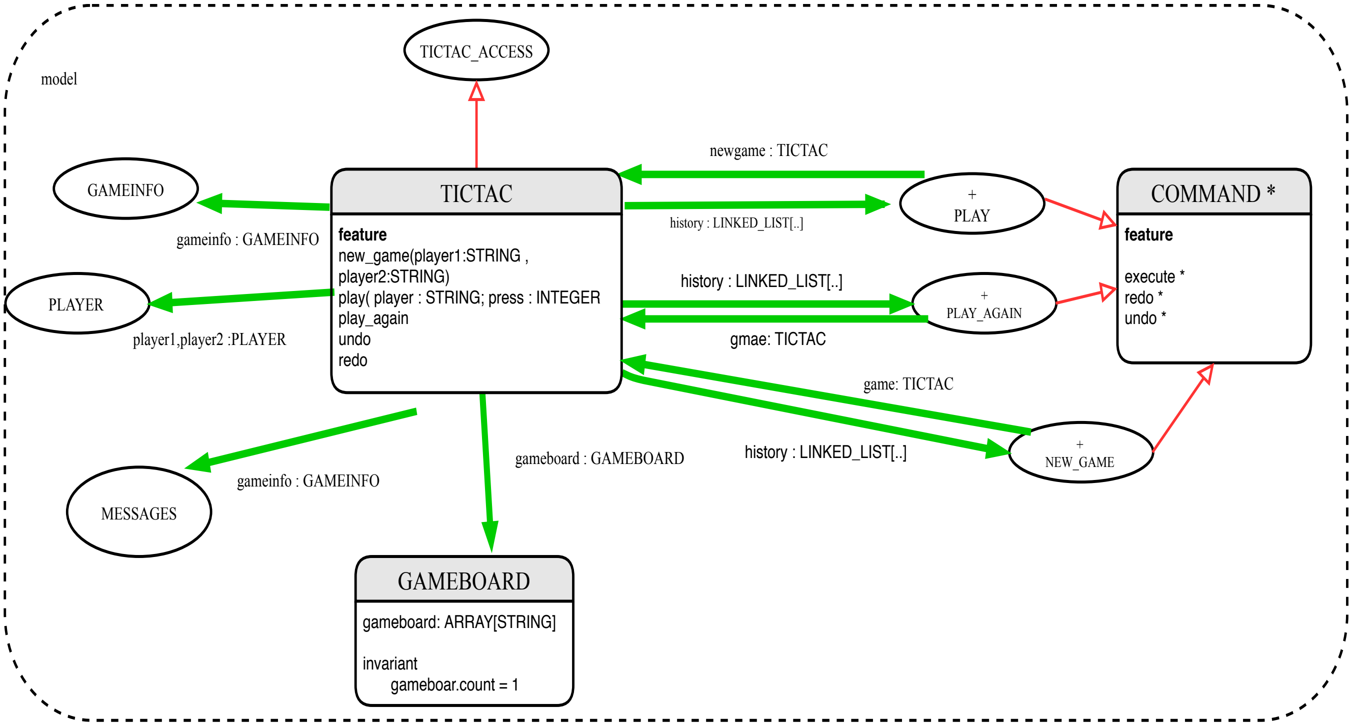
Name : gefei song
Signature: Gefei Song
Prism Account : arisgf

CONTENT:

Top level view:
This BON diagram includes classes: TICTAC(ETF_MODEL), TICTAC_ACCESS(ETF_MODEL_ACCESS), MESSAGES, GAMEINFO, PLAYER, GAMEBOARD, and a deferred class COMMAND which is inherited by classes PLAY, PLAY_AGAIN, and NEW_GAME. TICTAC class contains a LINKED_LIST[COMMAND], which is called “history”, can store past user commands, and undo/redo mechanism can be done by using this list.
TICTAC class contains all the necessary tictactoe game information such as players and game board instances, game status information and all the user commands which can be called from abstract_ui, if one of these commands be called, it will create a corresponding object (from class PLAY, PLAY_AGAIN, AND NEW_GAME) and put this object into the list.



Class description:

CLASS	DESCRIPTION	DESIGN DECISION
TICTAC	TICTAC is the class that interact with abstract_ui. Including features such as new_game, play, play_again, undo, redo. We can create corresponding objects in these features.	Include all the necessary game information. After creating an object of this class will initialize the tictactoe game and create a game board, two players, use history : LINKED_LIST[COMMAND] to store user command with polymorphism and dynamic binding
COMMAND	Deferred class include deferred features such as execute, undo, redo.	We need a deferred class to implement polymorphism and dynamic binding when we want to store commands' objects into history list.
NEW_GAME	Inherit COMMAND class, create an instance of this class will start a new tictactoe game, and clear undo/redo list, create two new players, can not use undo/redo command on this class.	Create a new game
GAMEBOARD	A tictac game board will be created when call this class.	We need a game board instance to store relative game board attributes, use an ARRAY[INTEGER] to represent a tictac game board, this array has 9 elements which represent nine position of game board.
PLAY	Inherit COMMAND class, create instance of player's operation, can implement undo/redo mechanism.	A play(name, operation) command object which can by stored in history list.
PLAY_AGAIN	Inherit COMMAND class, create a new game with same pair of players.	Reset game board and switch two players' play order.
PLAYER	Creating instance of a player include player's attributes: name, score.	Store players' attributes.
MESSAGES	Messages that will report wrong operations to players.	Once we create a TICTAC class object, we also create an object

	Each message is an attribute of this class.	of MESSAGES class in it.
GAMEINFO	Game instructions. Each instruction is an attribute of this class.	Once we create a TICTAC class object, we also create an object of GAMEINFO class in it.

Detect a winning game:
I create an array of string in class GAMEBOARD called “gameboard” with 9 elements, which are filled with string “_”, after each “play(string: name, integer: position)” command is executed, string “_” in array[position] will be replaced by string “X” or “O”, depends on which player. Then we can check if (array[1][2][3] or array[4][5][6] or array[7][8][9] or array[1][4][7] or array[2][5][8] or array[3][6][9] or array[1][5][9] or array[3][5][7]) equals “XXX” or “OOO”. If it equals, one of player win the game.

Undo/redo design :
In the TICTAC class (ETF_MODEL) to create a LINKED-LIST[COMMAND] called “history” to store COMMAND instance after executing “execute” command. We can implement undo/redo command by moving cursor back and forth to get different old and executed command objects.

Undo/redo mechanism can be used in PLAY command.

After creating a PLAY object, we can put it at the end of history list. Before store this object ,we must check the cursor position and remove all the COMMAND instances which are at the right of the cursor to make sure undo/and redo mechanism run correctly. After put it at the end of the list, we should move cursor forth to make sure the cursor point to the right position.

After we call NEW_GAME command, history list will be cleared, every command object will be deleted and we can not call undo/redo until we execute PLAY command in the new game.