1a.

Each state for Tic-Tac-Toe would correspond to a 3x3 grid filled with some number of X’s and some number of O’s. For example, at the start of a game, the state would correspond to an empty 3x3 grid. The current state of the game affects the actions available because moves cannot be made in filled squares; this allows for a greater or lesser number of choices depending on how many squares are filled.

1b.

I would give +100 for a win, -100 for a loss, and 0 for a draw. This incentivizes the player to win as much as it disincentivizes the player to lose. For actions that do not end the game, I would give a reward value of 0 to actions that do not end a game. This is because there should be no incentive to win as quickly or slowly as possible; I want to train the model to do its best to win regardless of game length.

1c.

The environment should give rewards after the opponent plays, since the new state for the agent’s next move is defined by both the agent’s previous move *and* the opponent’s response.