

Agenda

→ class diagram ⇒ Tic Tac Toe

→ Find winner in $O(1)$

→ Coding the requirements

→ Models

→ Basic functionality

}

CLASS DIAGRAM

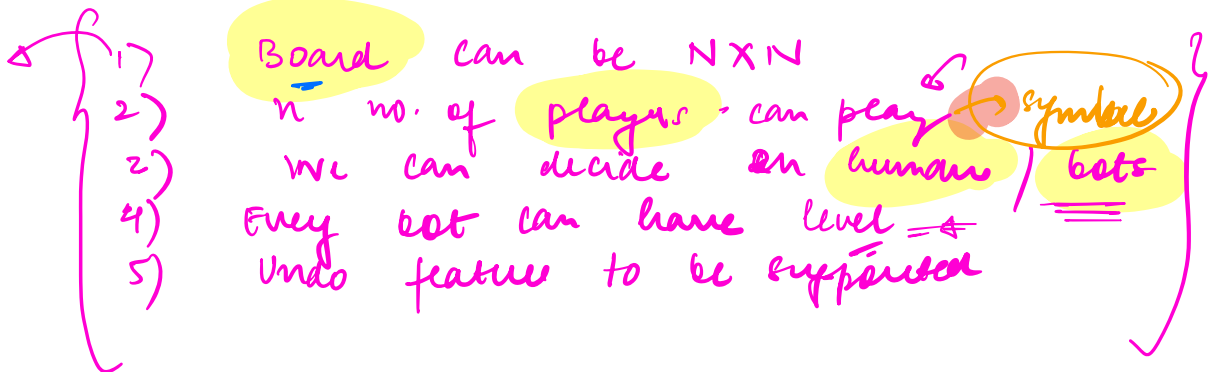
→ visualisation

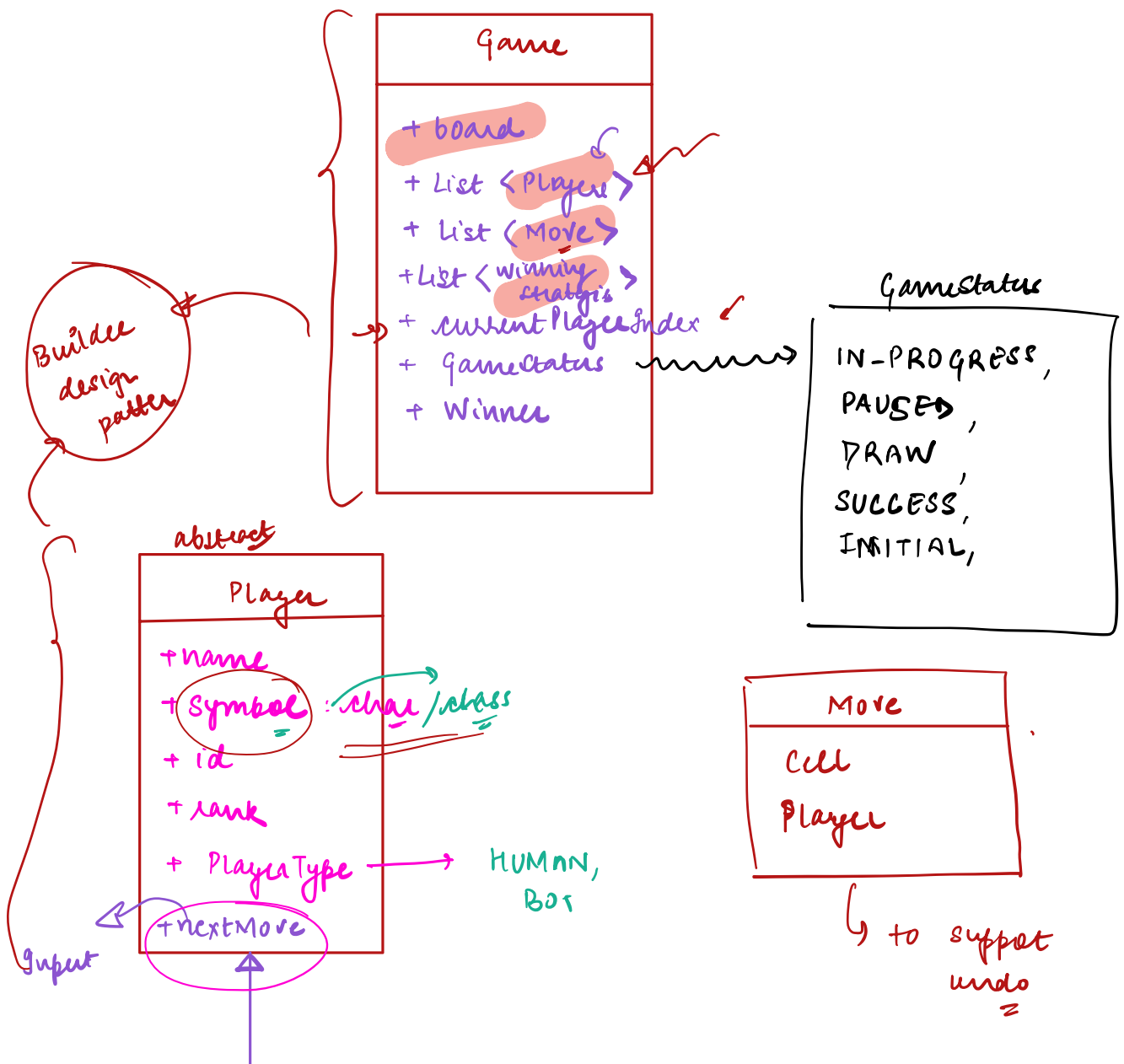
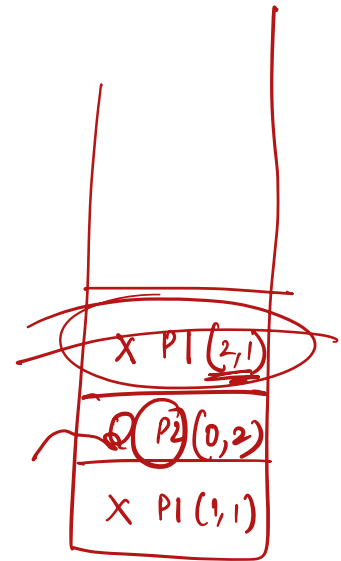
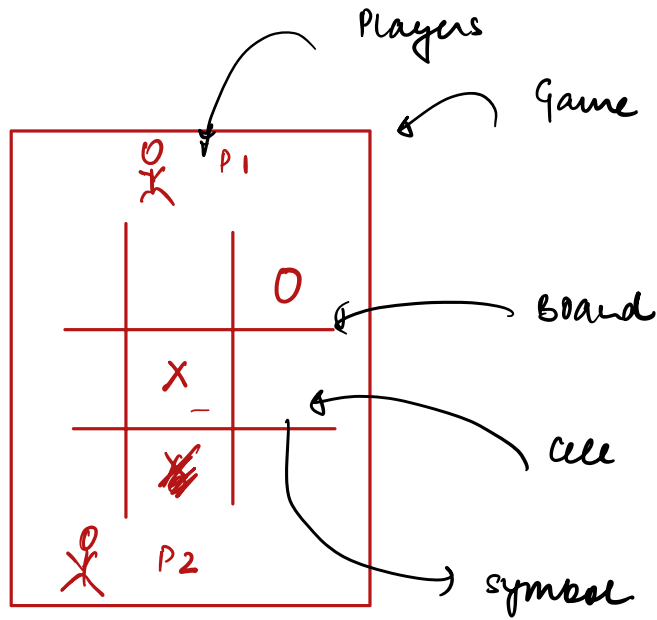
→ Nouns

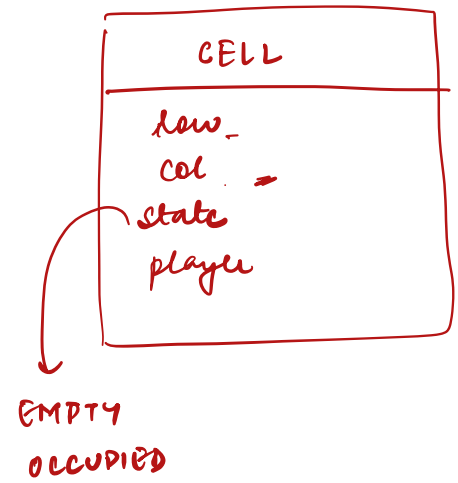
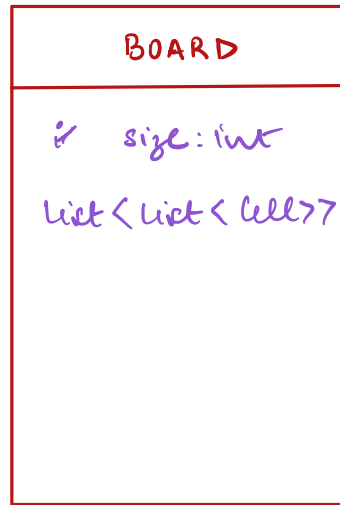
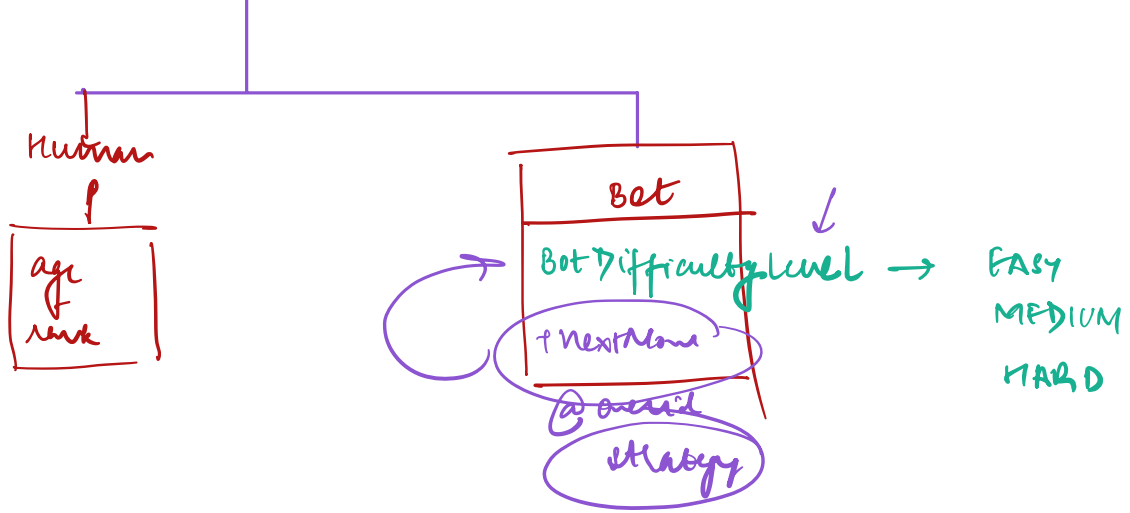


→ Entities

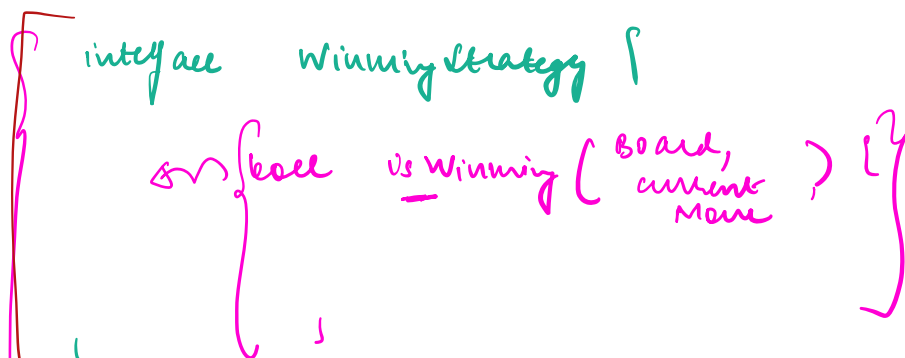
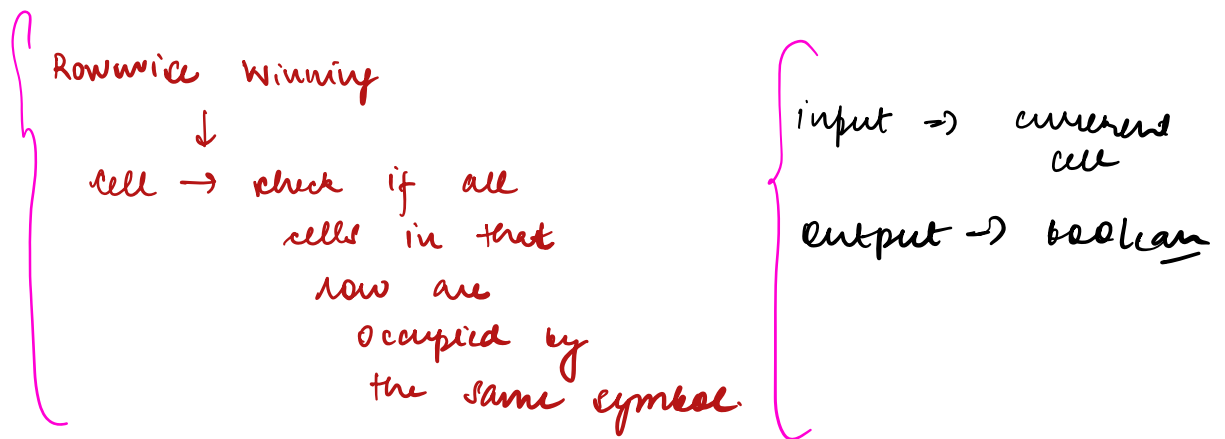
→ Design patterns.



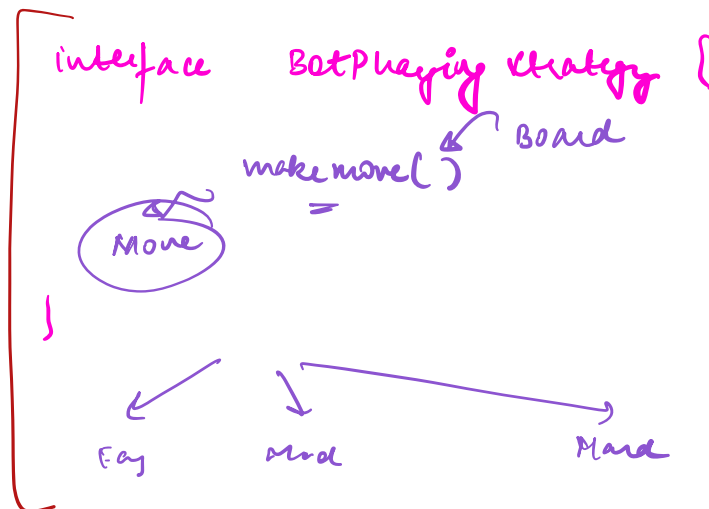




O	X	E	
X	O	E	E
	X	O	
			O



Strategy Design pattern



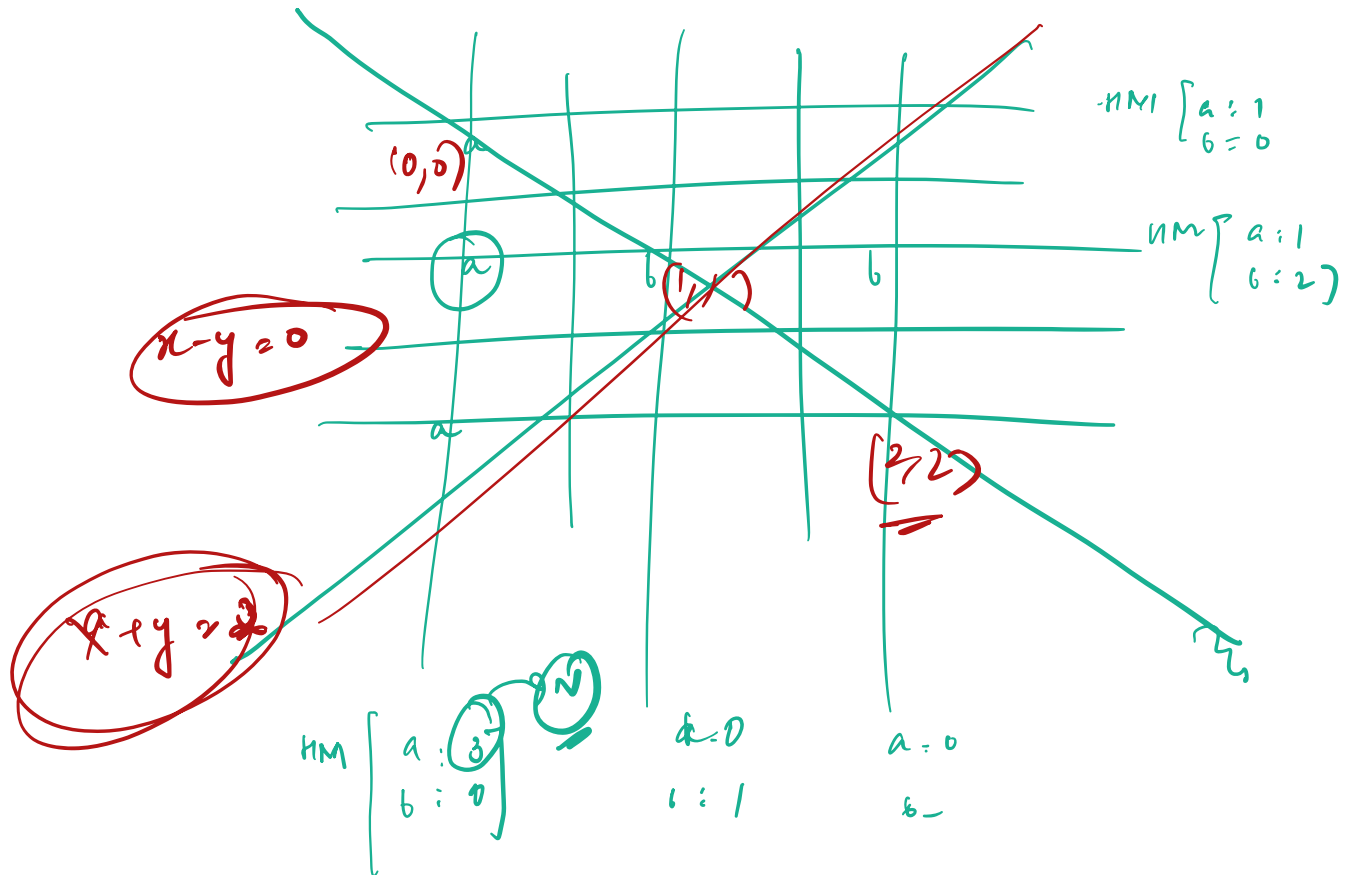
Bot status

get a particular playing strategy

Simple factory

BUILDER
FACTORY
STRATEGY

bool checkwin (board , cell) $\Rightarrow O(1)$



class WinningStrategy {

HashMap < Symbol, int > row [N]

HashMap _____ col [N]

diag ~~[N]~~

anti ~~[N]~~

checkwin (cell , board)

row [cell.row] [cell.player.symbol]++

col [_____ col _____]

valid Sudoku.

~~def~~ [
 if (~~var~~ == N)
 return true

MVC

Break \Rightarrow 50 mins.

10:20 pm \rightarrow coding

Models
{
 Controller
 Service
 }

Main - java