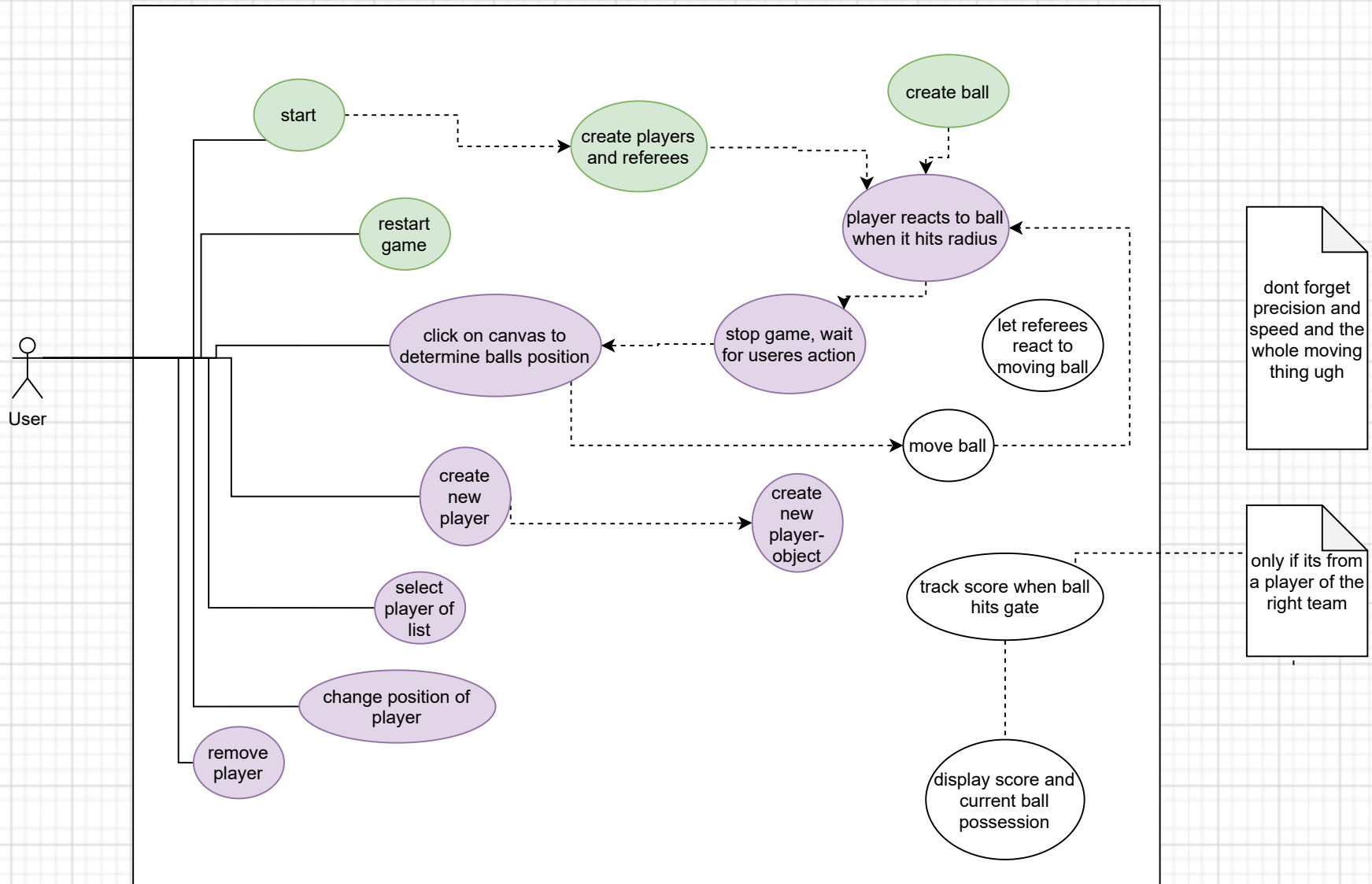
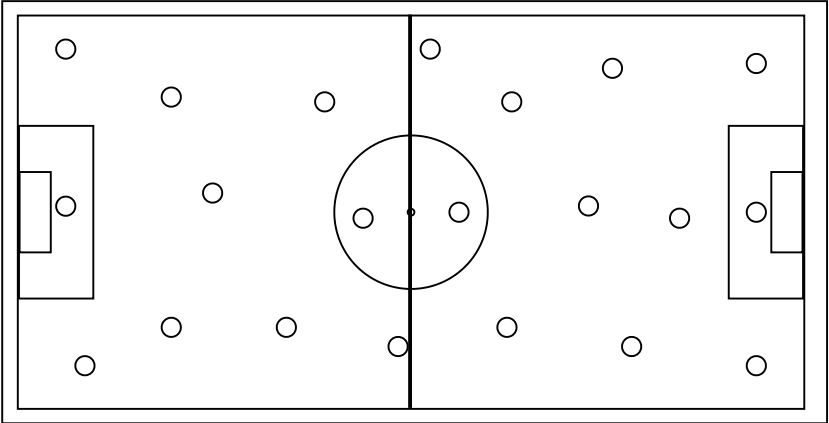


footballSimulation: Use-Case





current ball  
possession:  
scoreline: x:y

team settings

Team A:

Team A Colour:

Team B:

Team B Colour:

reset

start game

speed of player, max,  
min

precision of player,  
max. min

player settings:  
player: Player XY  
change shirt  
number of player:

reset

submit

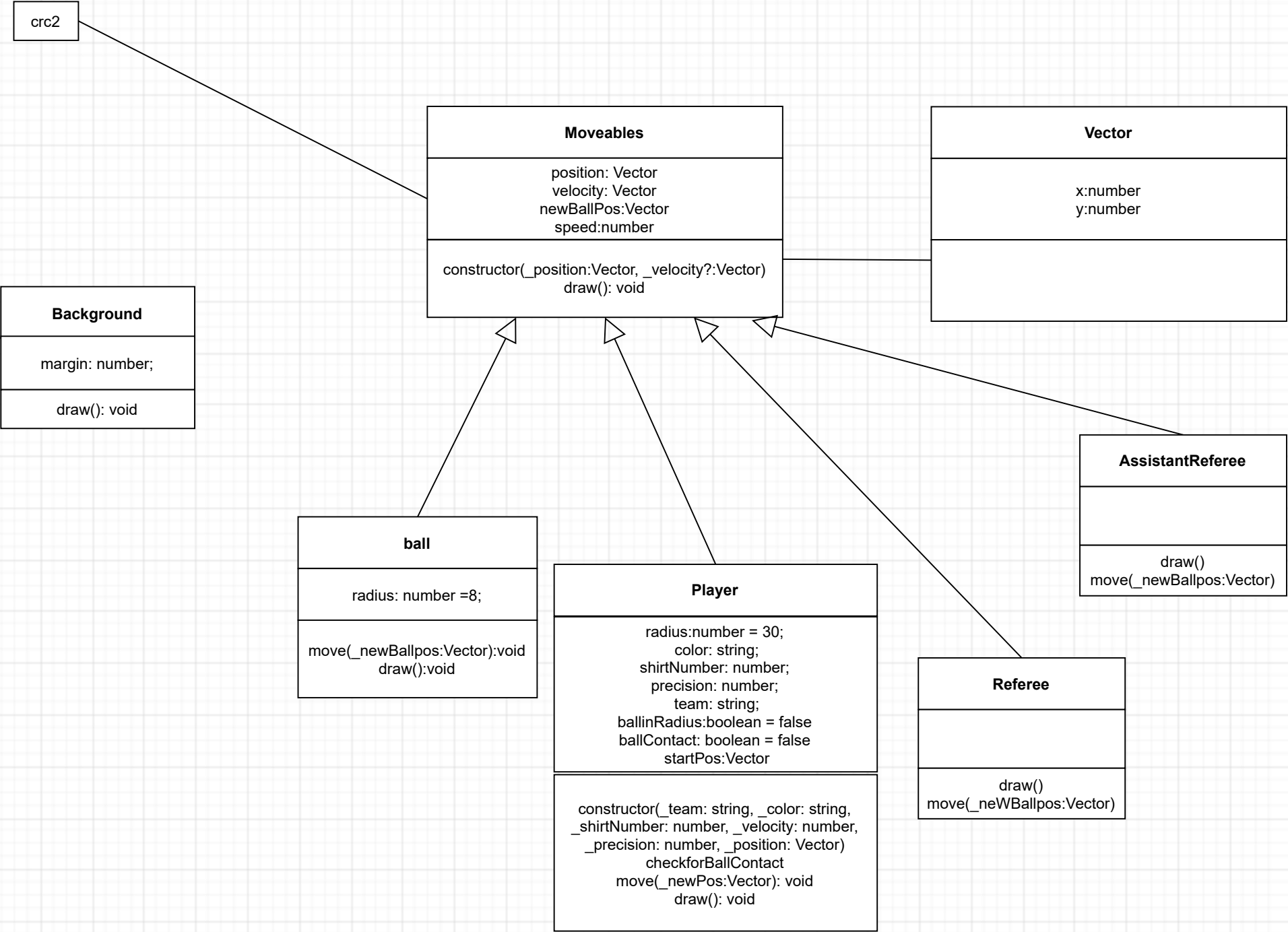
create new player  
for Team A:

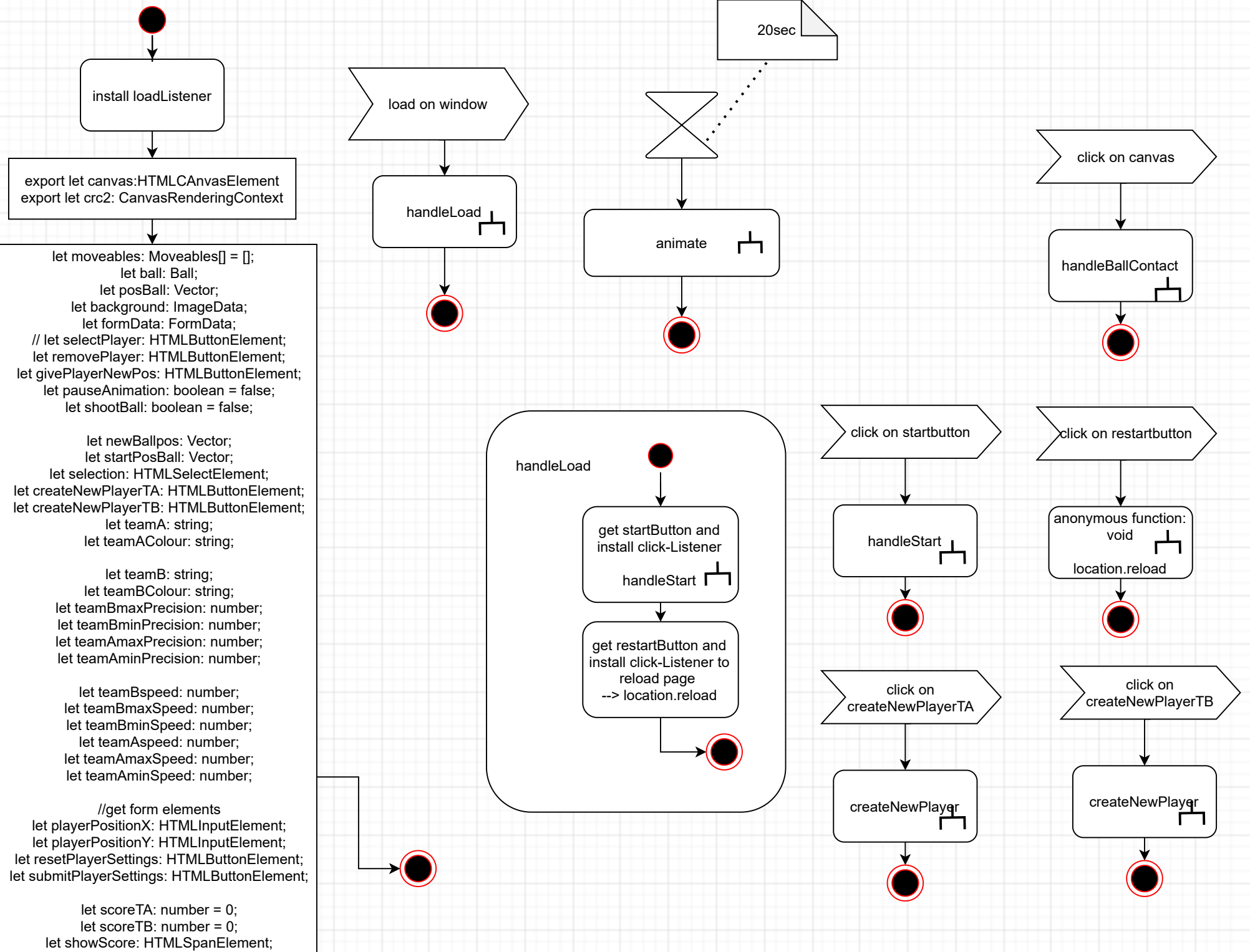
create new player  
for Team B

restart game

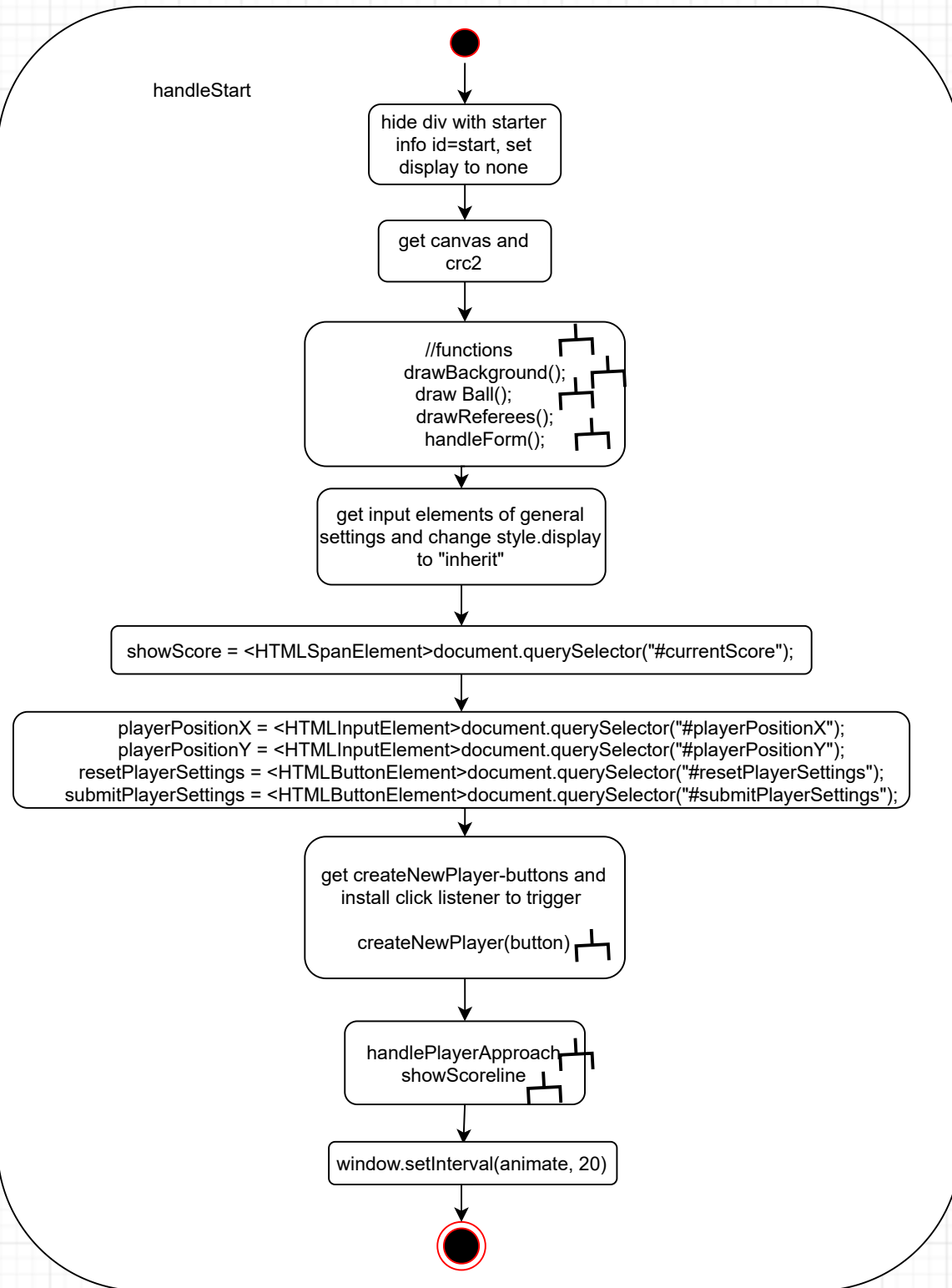
start game

footballSimulation: Class Diagram

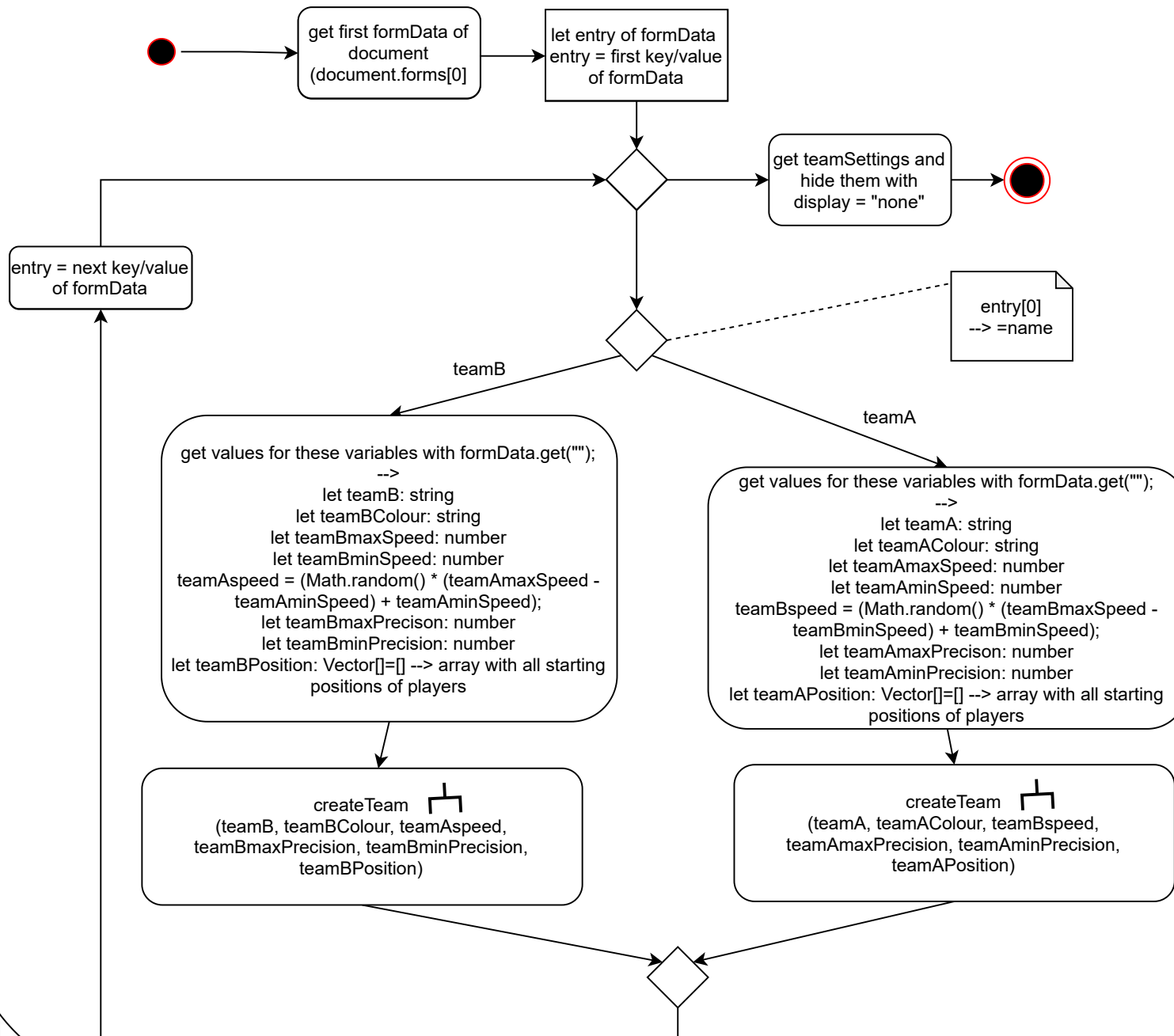




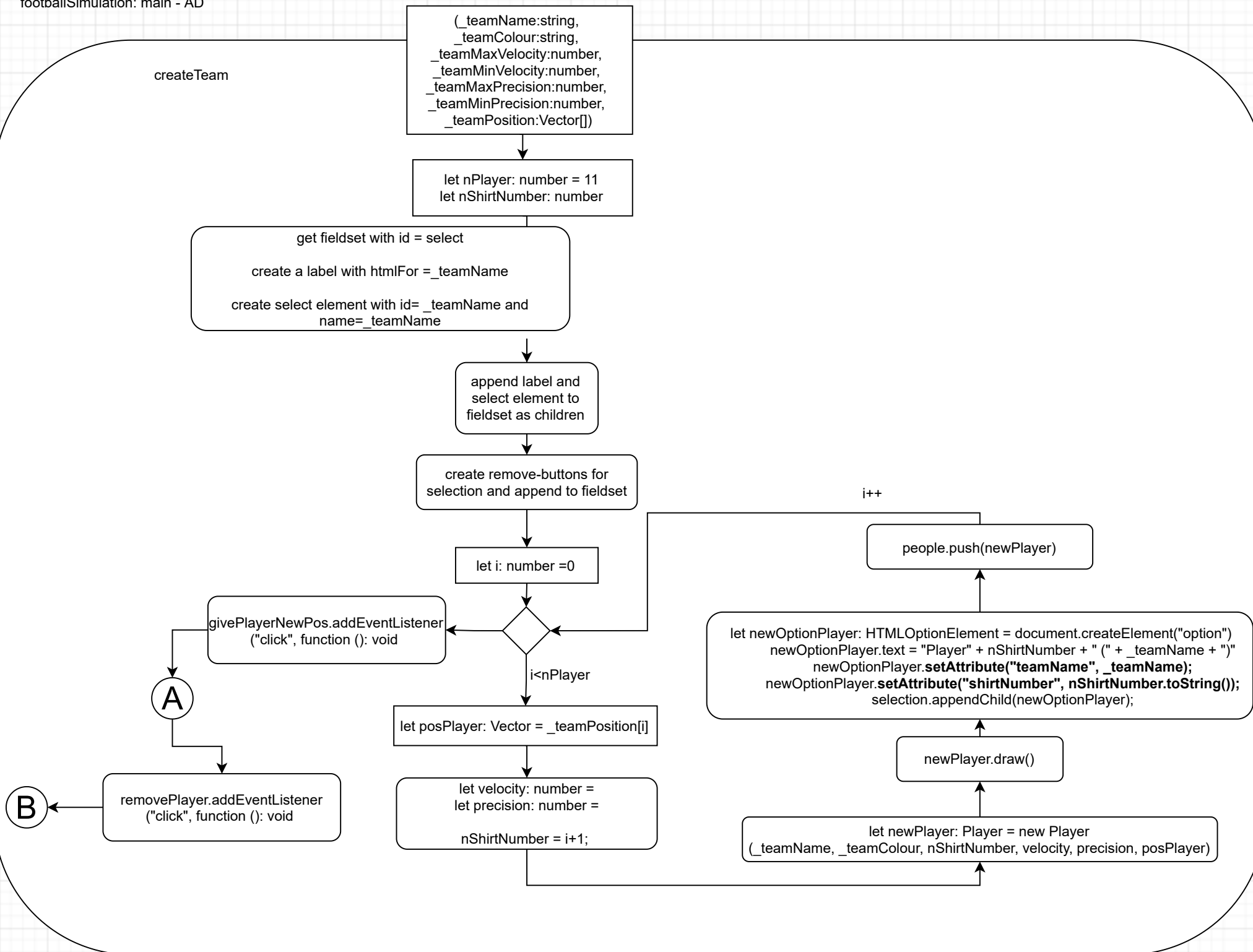
handleStart



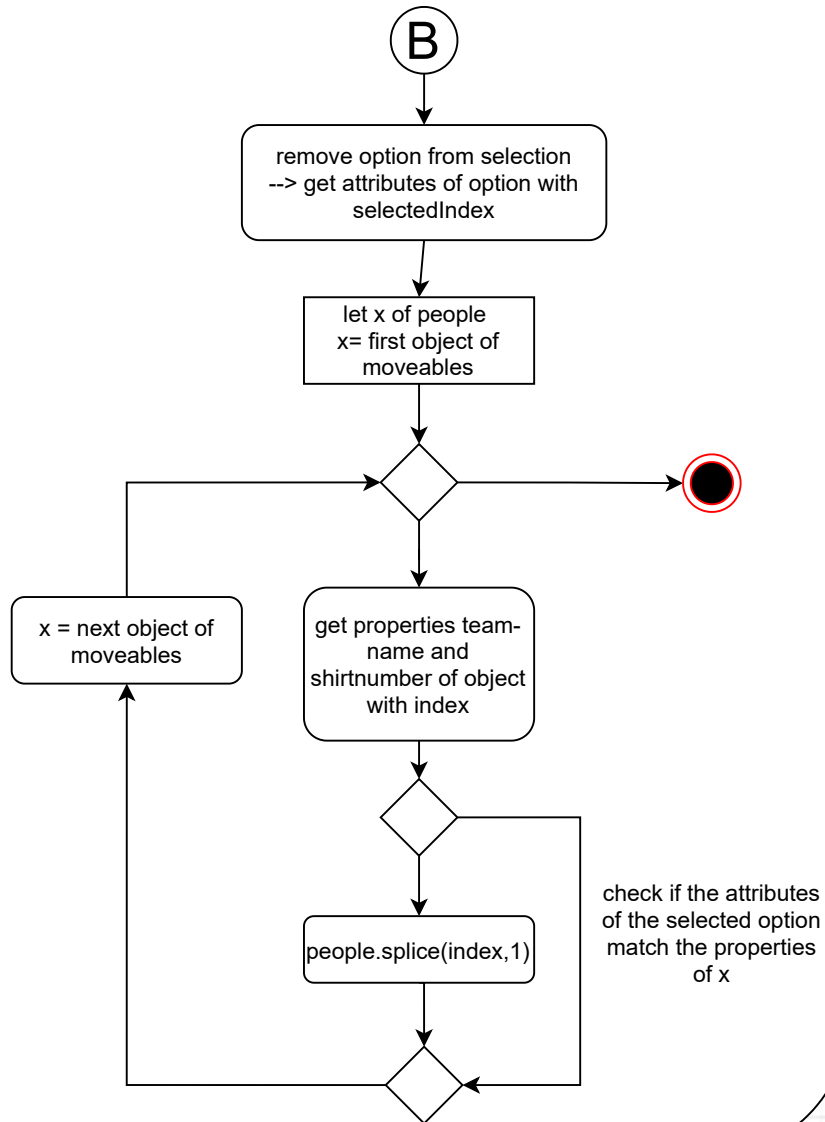
handleForm



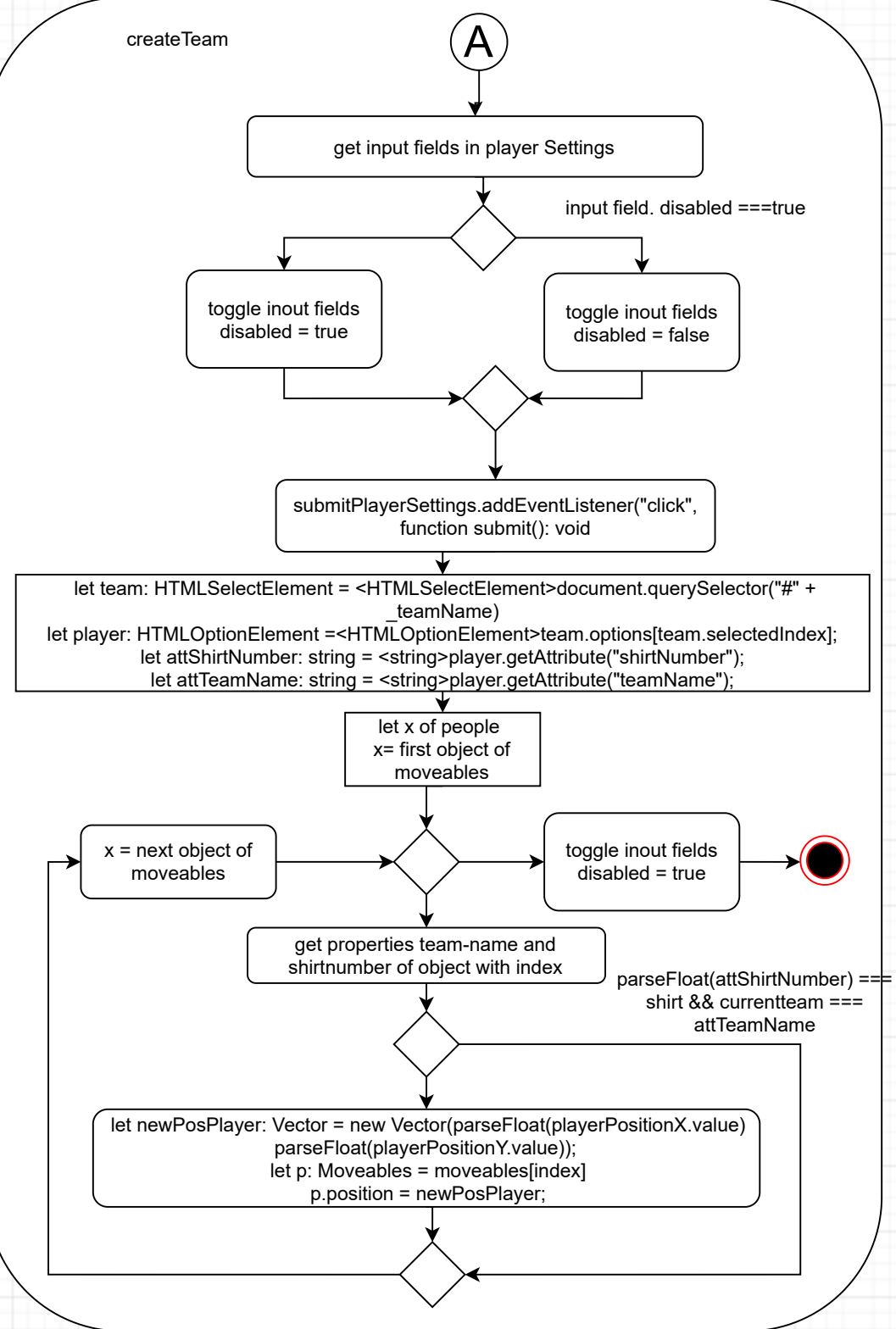
createTeam



# createTeam

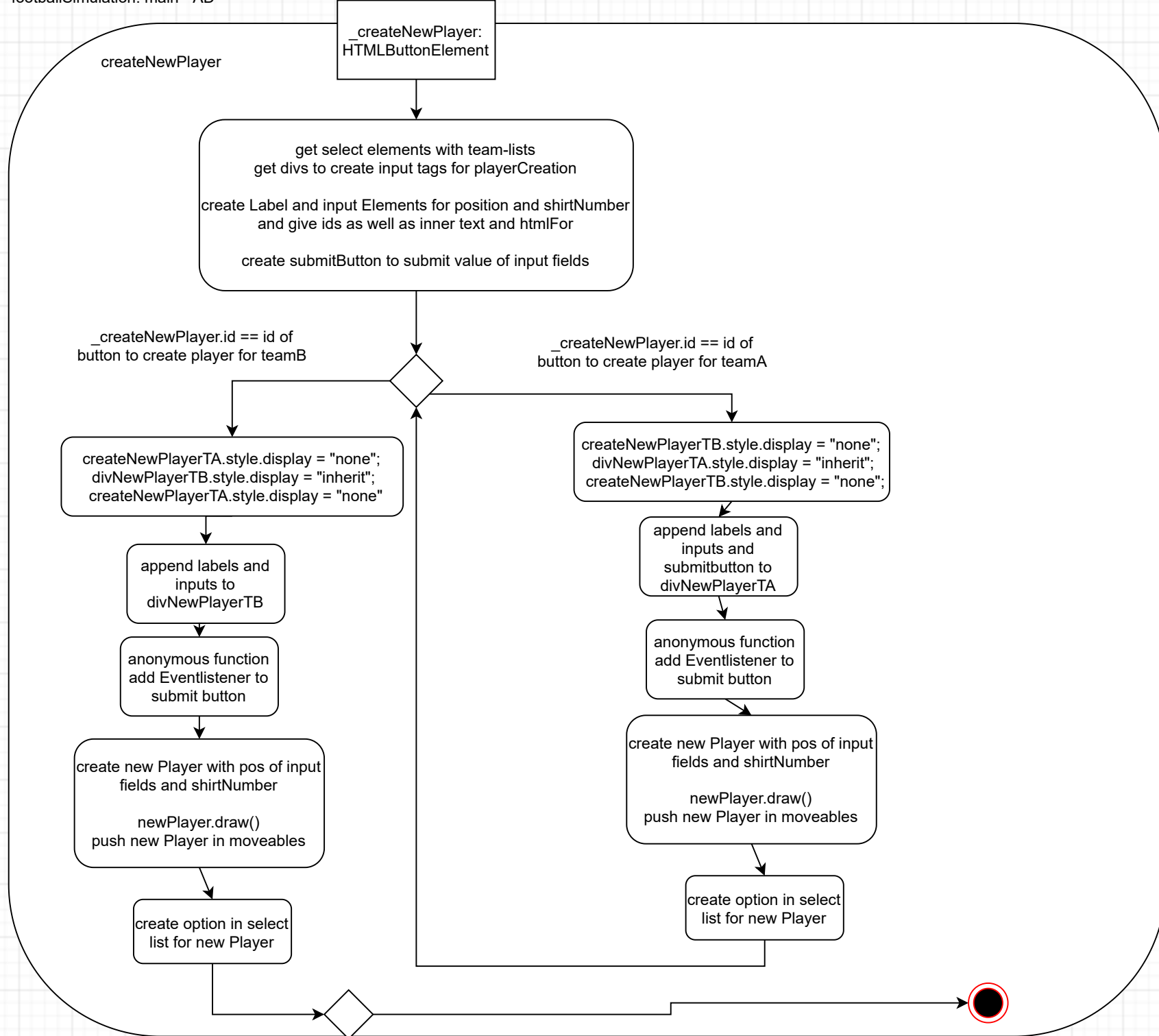


# createTeam

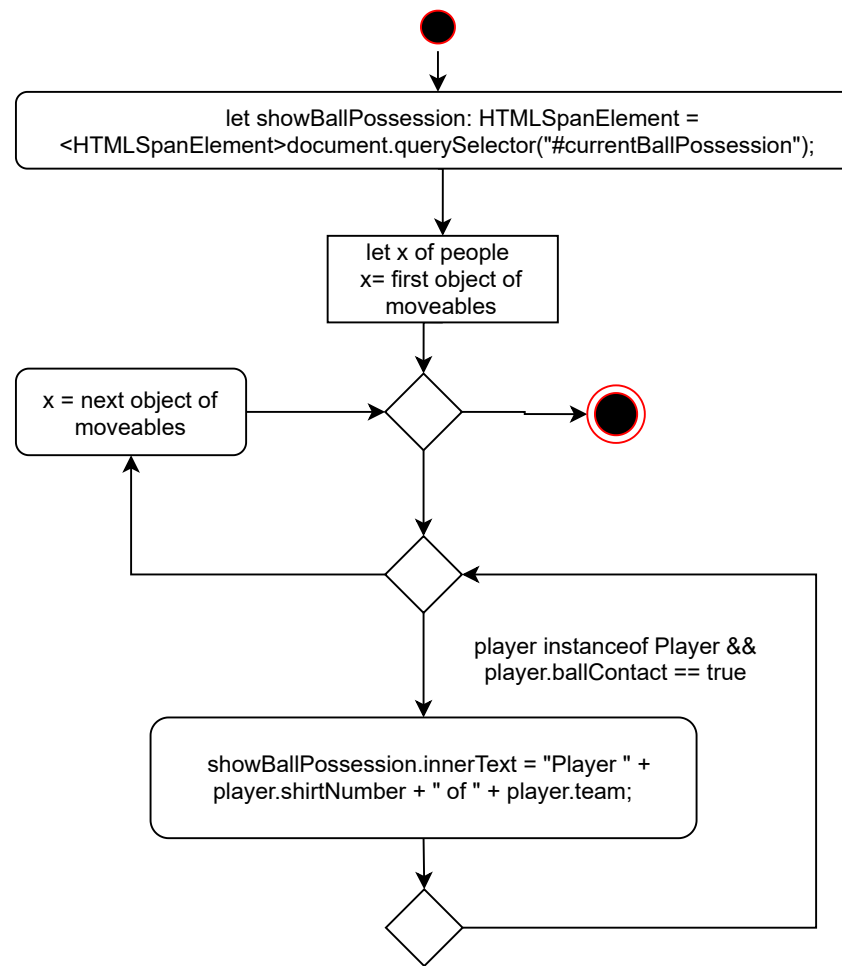


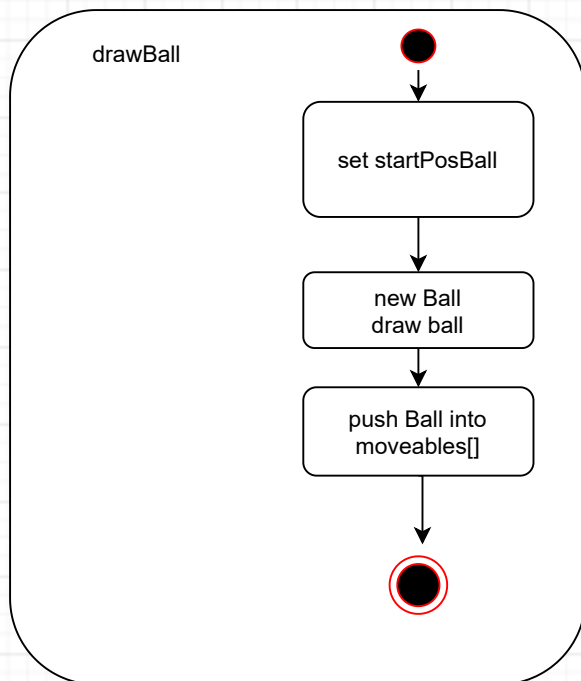
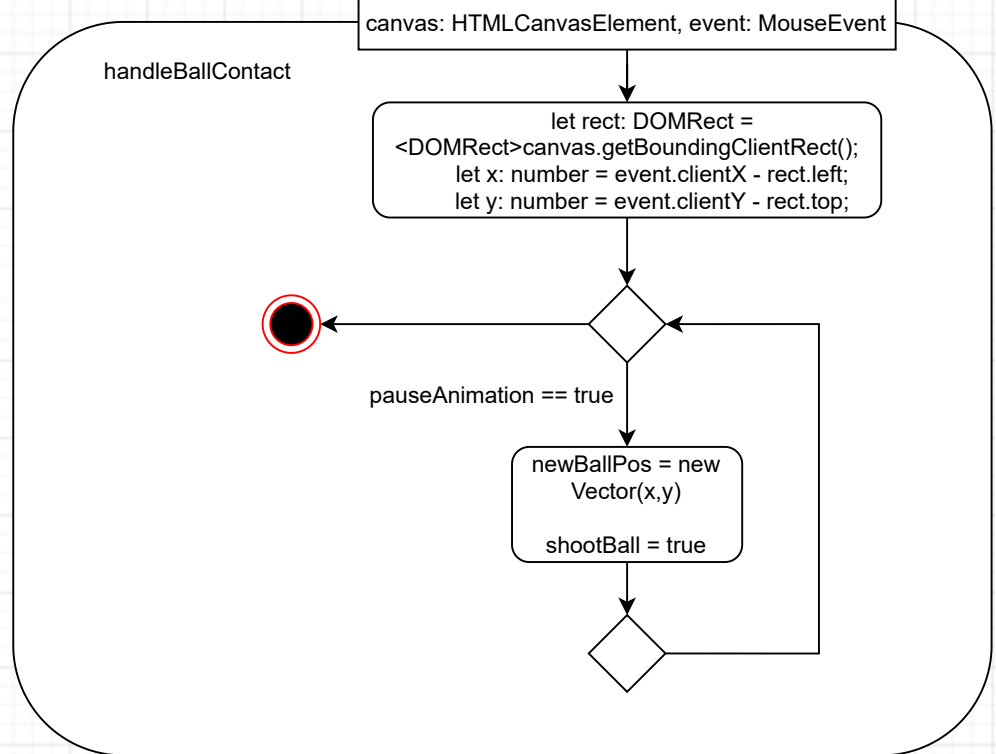
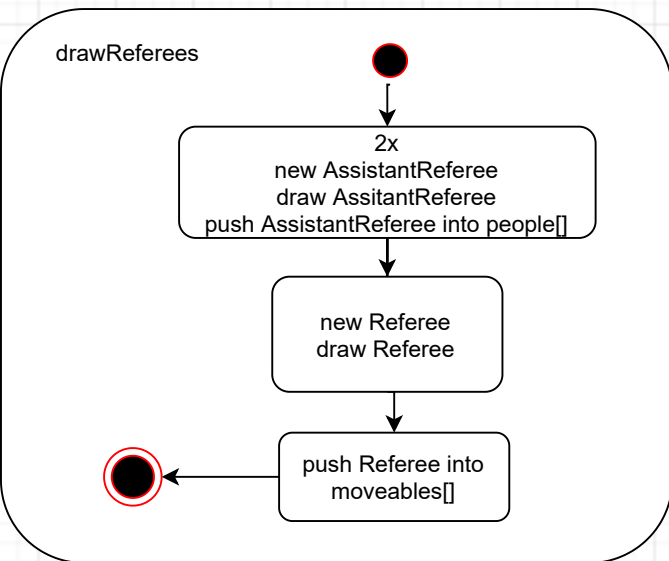
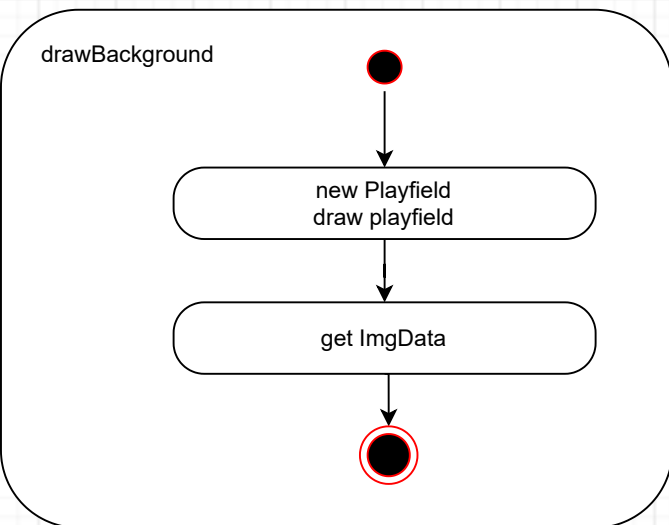


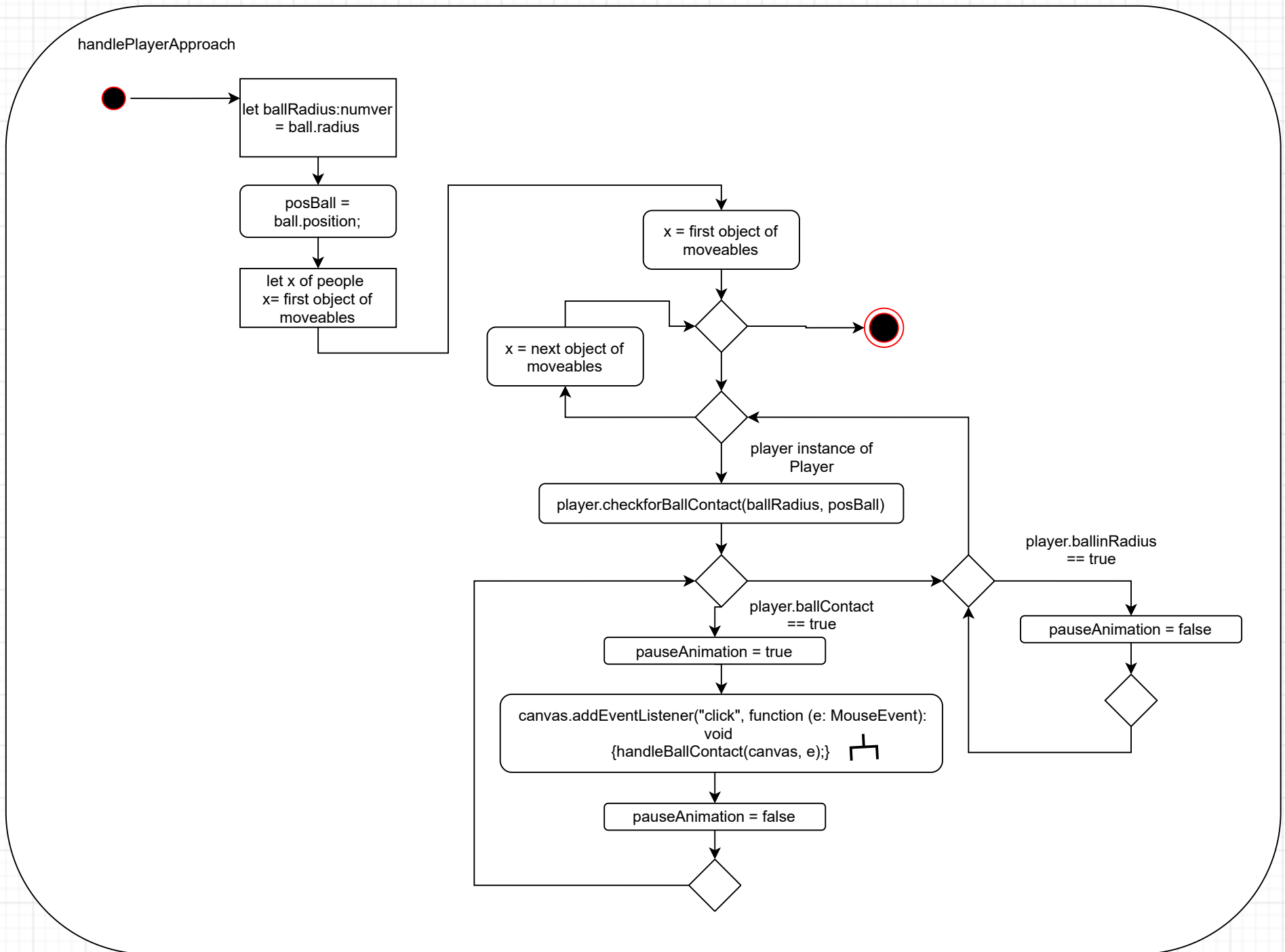


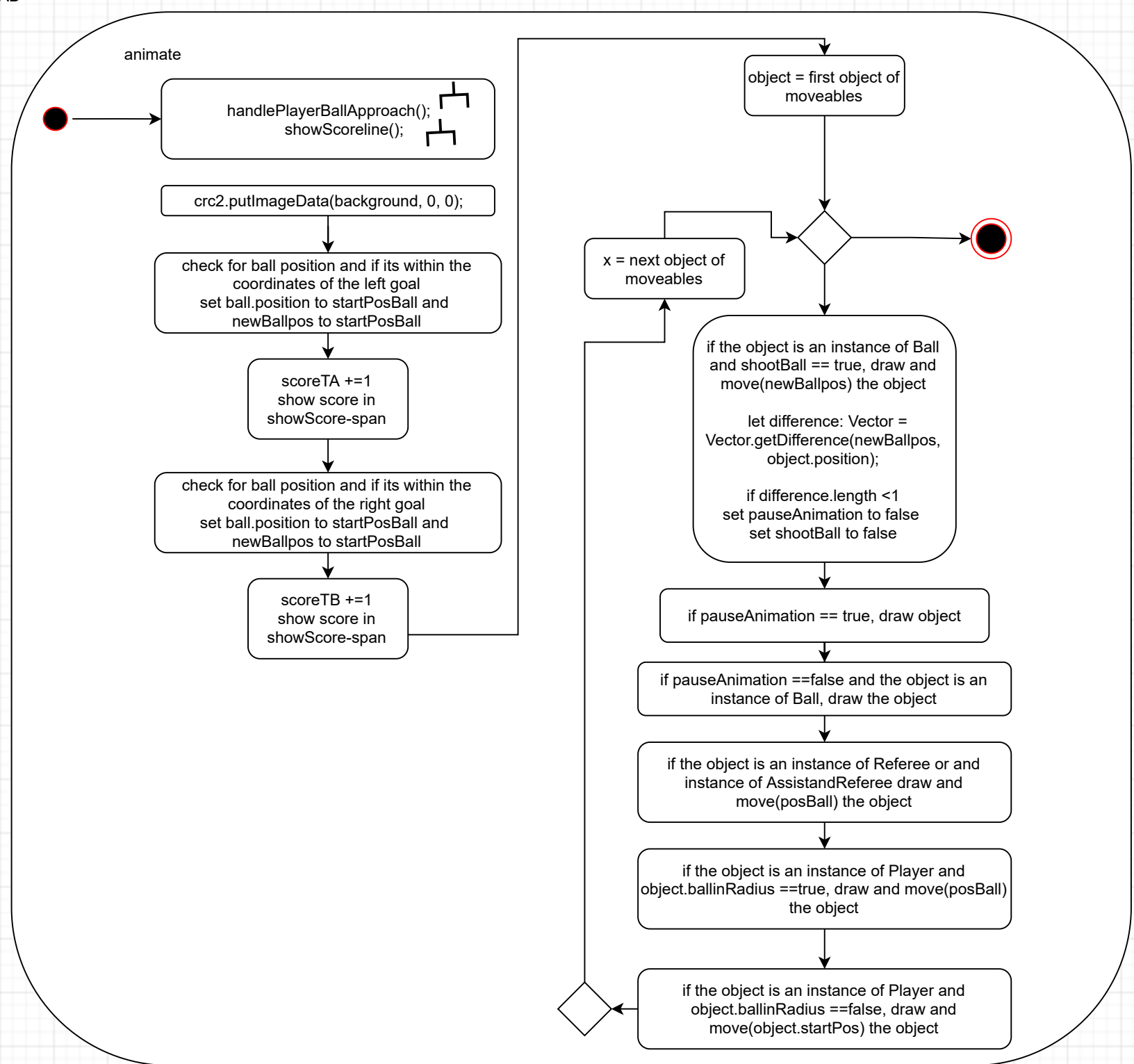


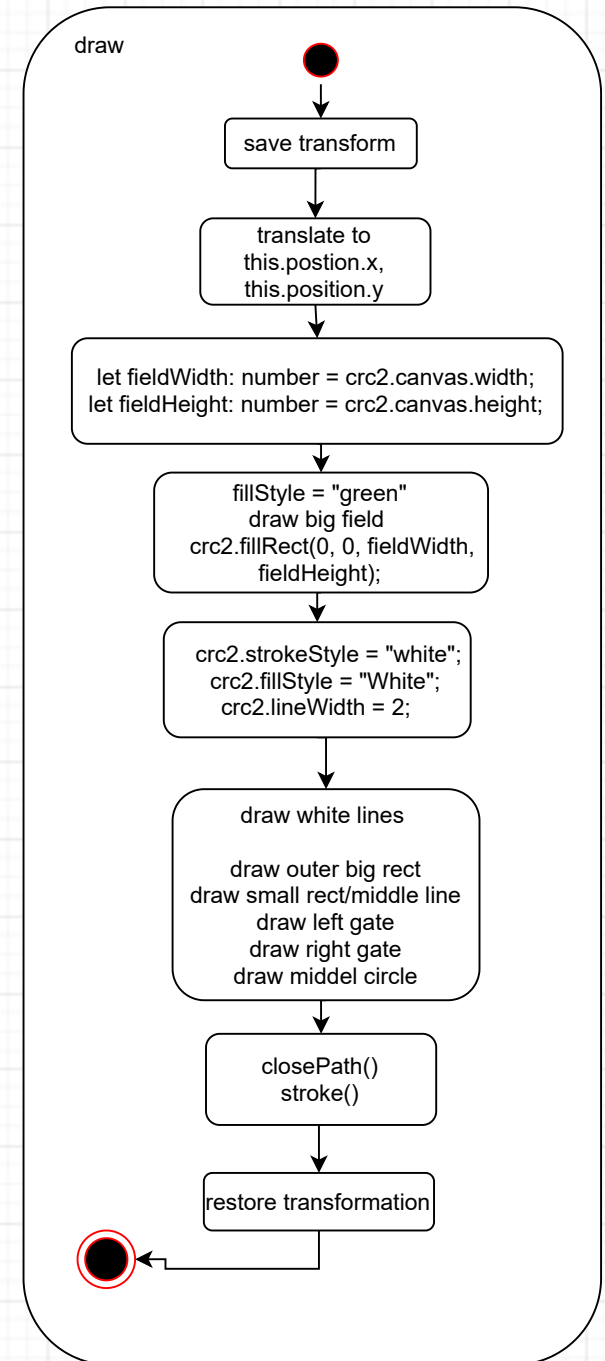
showScoreline



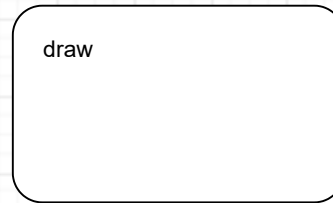
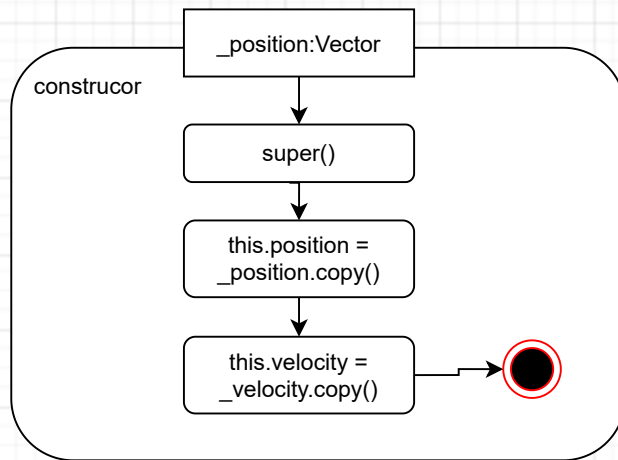




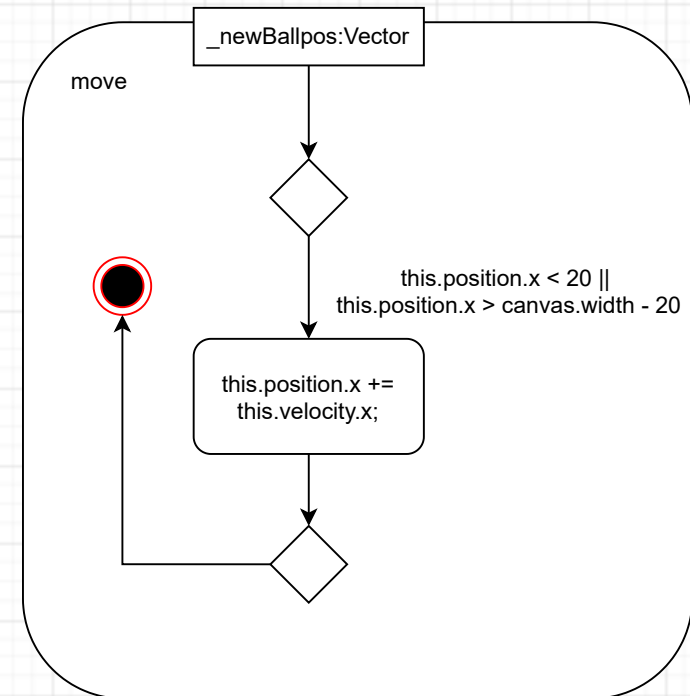
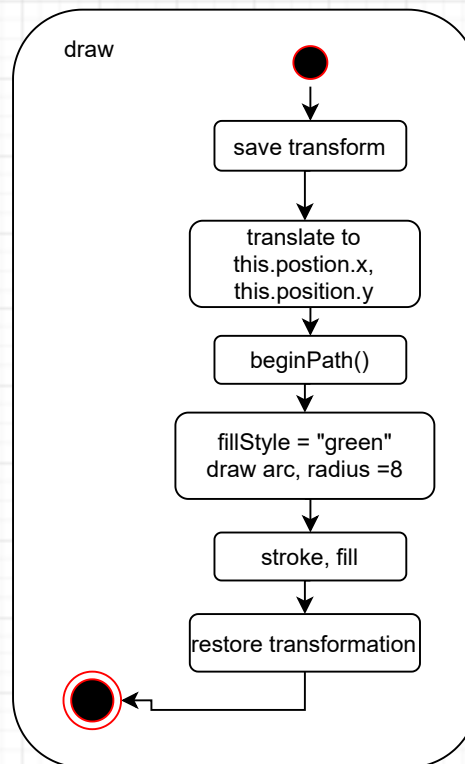




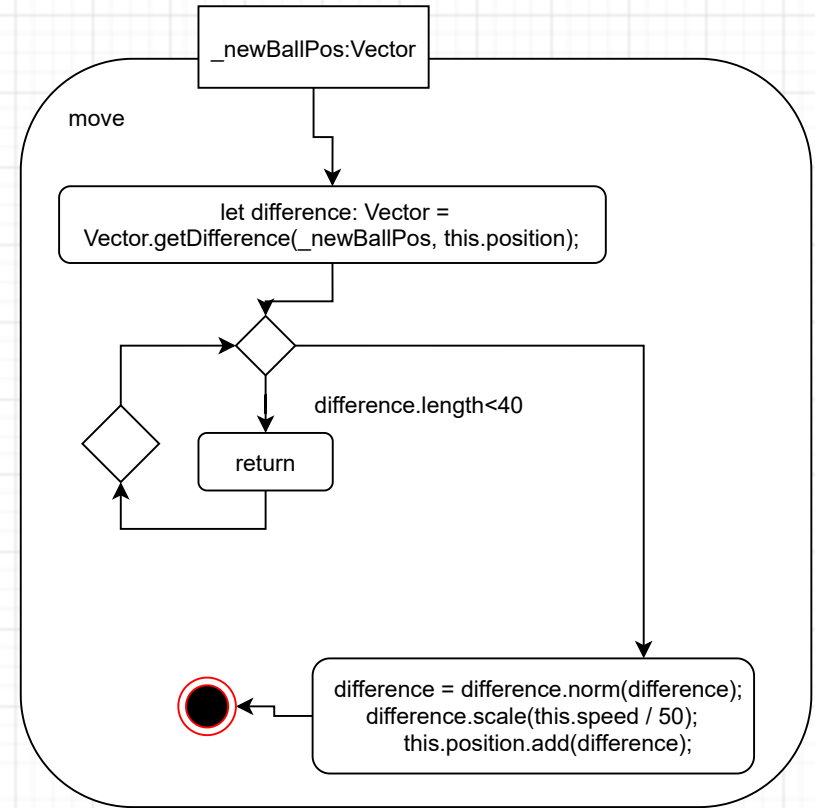
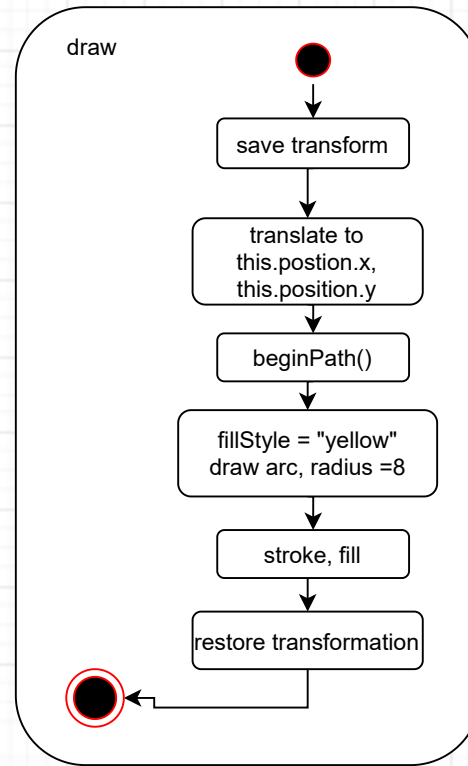
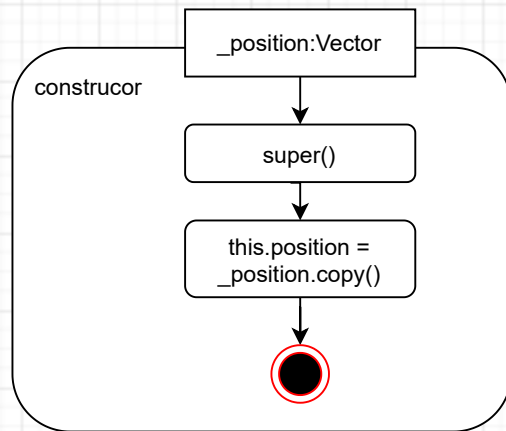
# footballSimulation: Moveable - AD

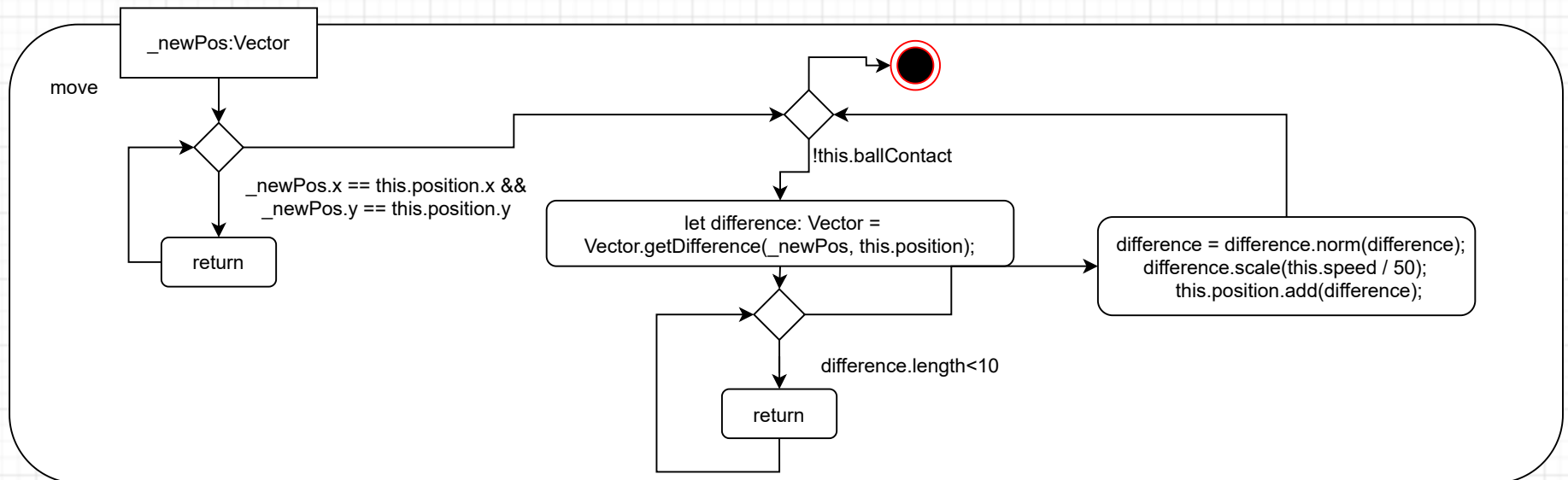
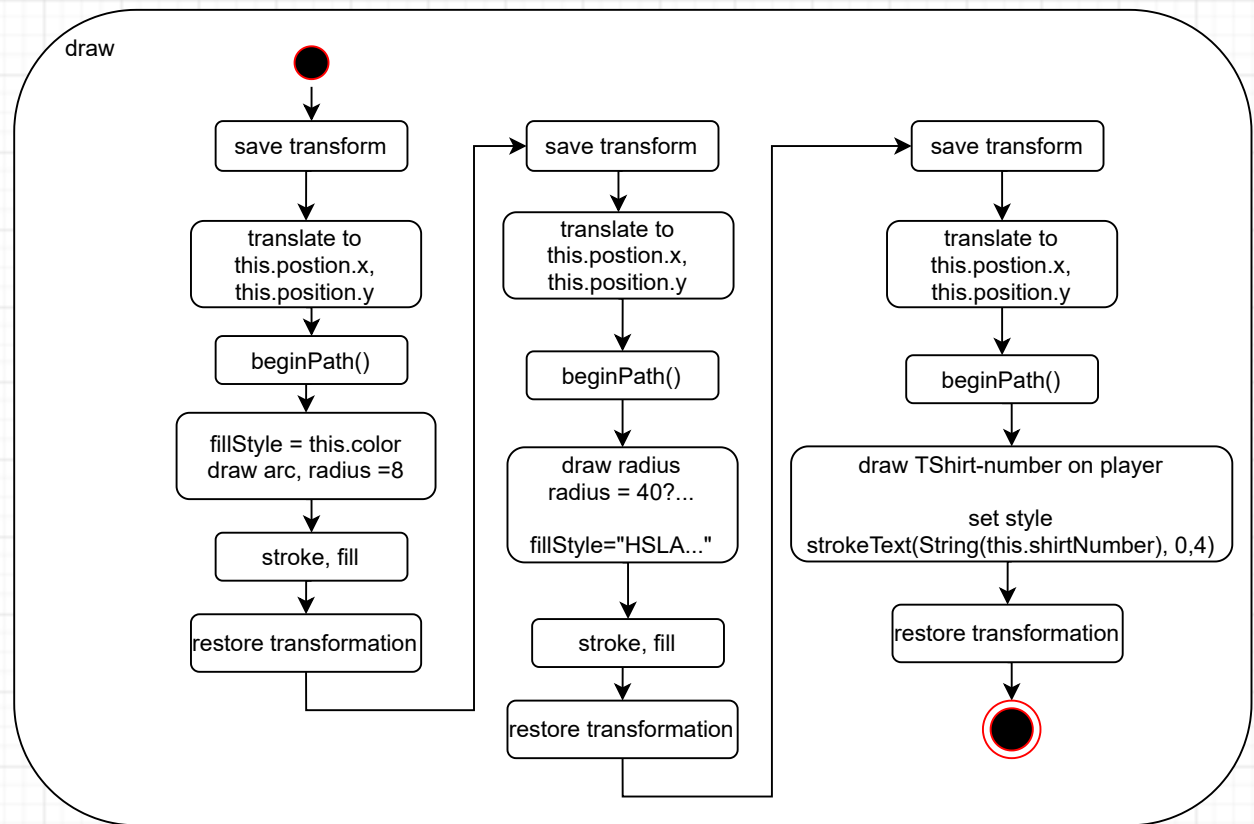
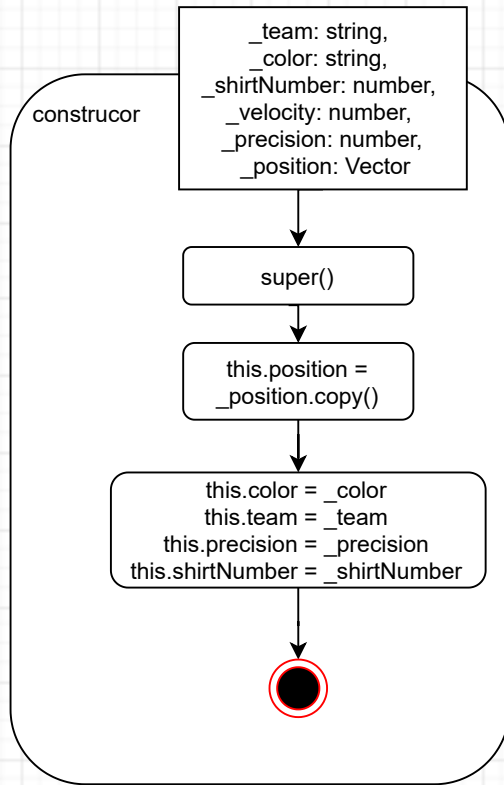


## footballSimulation: AssistantReferee - AD









checkforBallContact

\_radiusBall:number,  
\_poBall:Vector

let distX: number = this.position.x - \_posBall.x;  
let distY: number = this.position.y - \_posBall.y;  
  
let rSum: number = \_radiusBall + Player.radius;  
let distance: number = (distX \* distX) + (distY \* distY);

distance <= rSum \* rSum

this.ballinRadius = false

this.ballinRadius = true;

distance <= 150

this.ballContact = false

this.ballContact = true

