

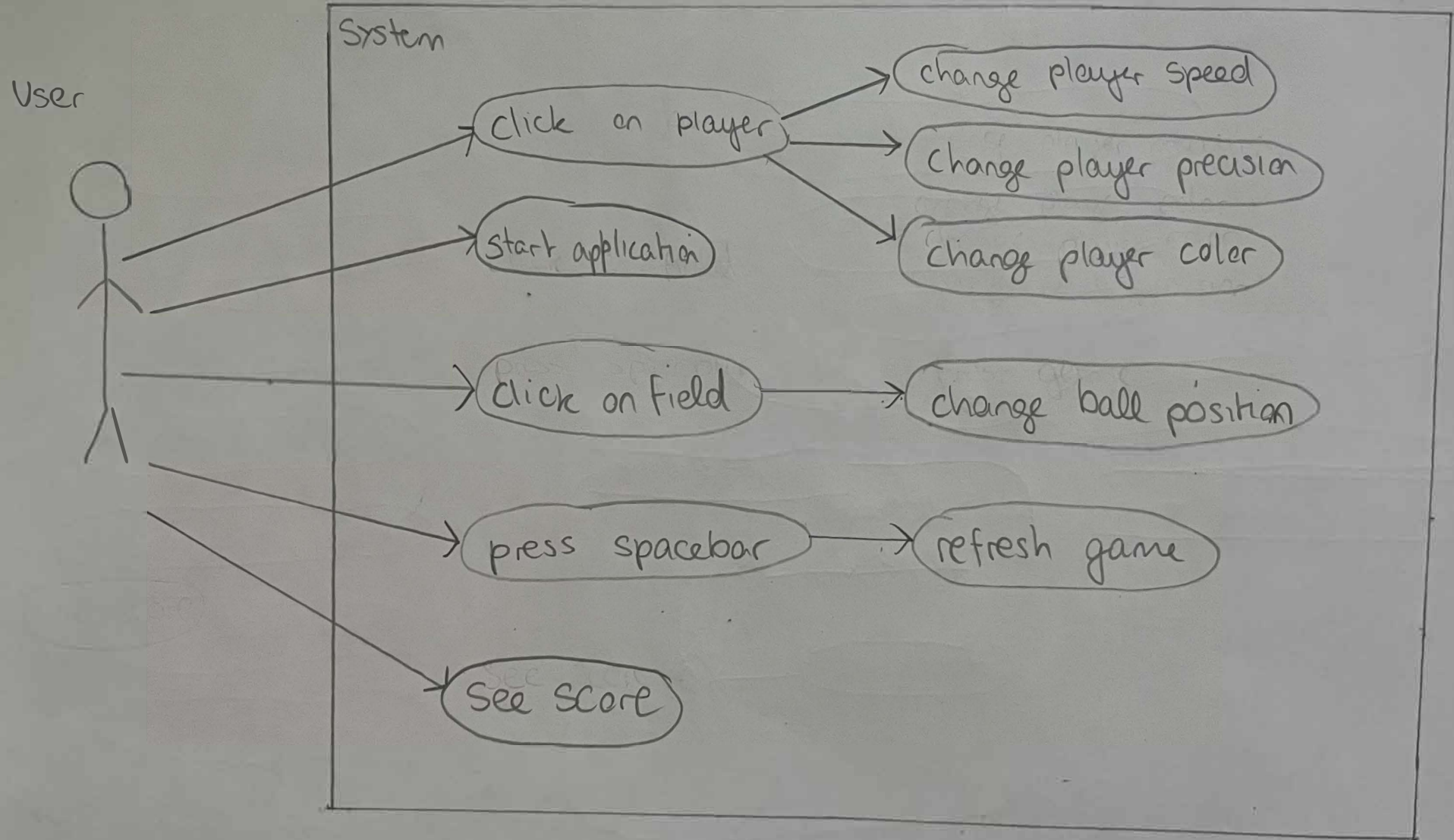
Endabgabe EIA2 Sose 2021

Interaktionsanleitung:

1. Zip-Datei runterladen und entpacken
2. Um das Projekt zu öffnen, Doppelklick auf die Html-Datei
3. Der Ball kann durch Klicken auf das Spielfeld bewegt werden
4. Um die Werte der Spieler zu ändern können die Slider benutzt werden
5. Um das Spiel erneut zu Beginnen, kann die Leertaste genutzt werden.
6. Wenn ein Tor erzielt wird, oder der Ball ins aus rollt, wird von der Mittellinie aus weiter gespielt.

Viel Spaß!

Use-Case Fußballsimulation



Konzept - Fußballsimulation

Desktop Anwendung

Fußballsimulation v 1.1.

Spielstand 0:0

Am Ball: Spieler X

850

Customize: Spieler X

Speed

min

max

Precision

min

max

Jersey color:



Name: Spieler X

Number: 17

fieldset + form Elements

input speed

input Precision

} zwischen diesen Werten wird zufällig einer zugewiesen

input Color

Colorpicker ändert Farbe des Spielers

Anzeige von Name & Nummer des ausgewählten Spielers

canvas
width: 850px
height: 450px

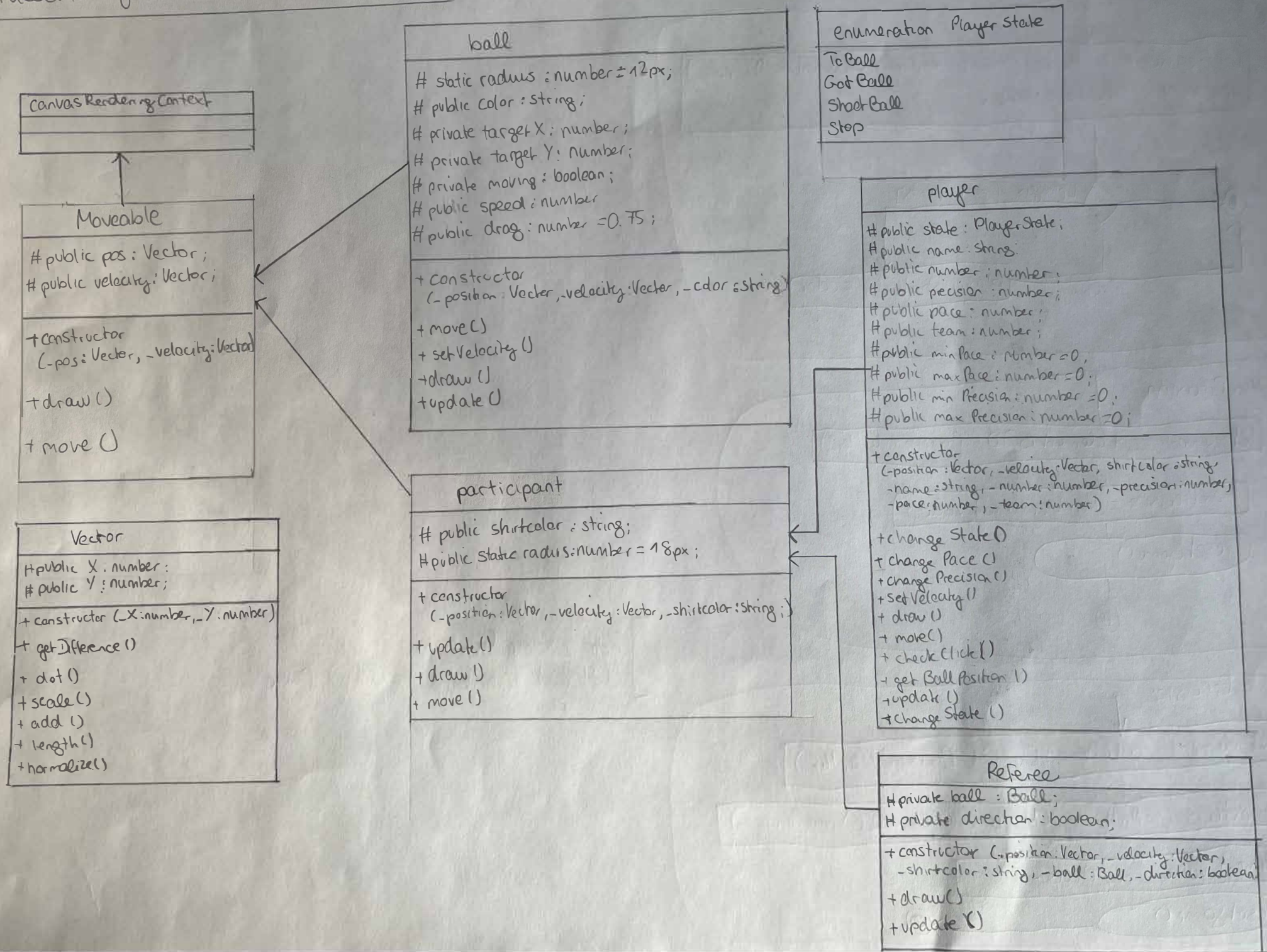
Ball
Folgt der Klickposition des Nutzers
radius: 12px

Schiedsrichter

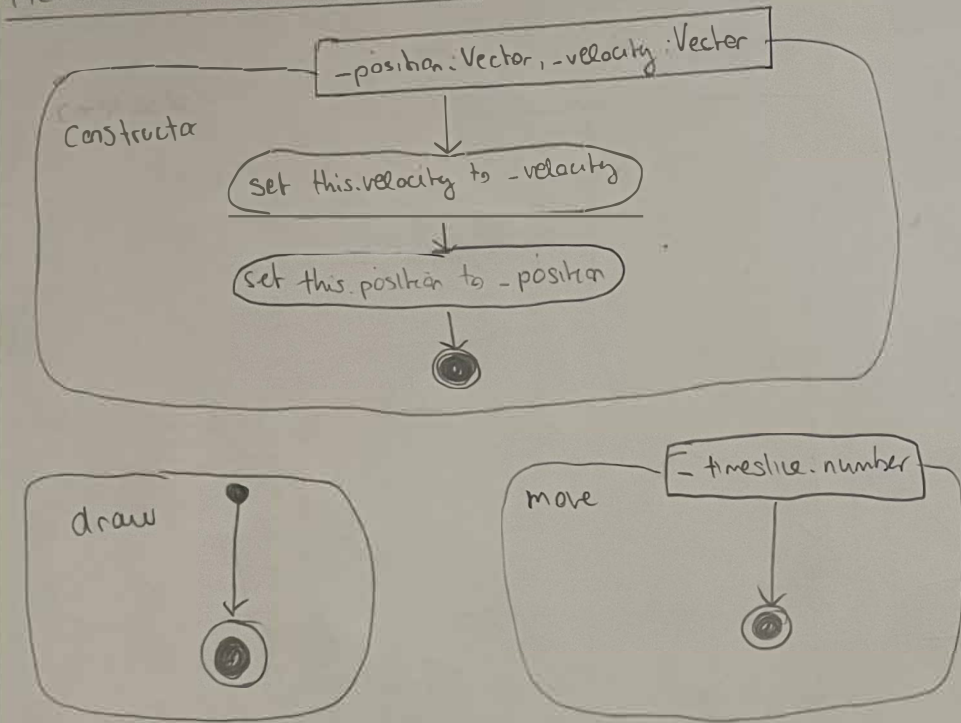
- 1: bewegt sich auf X-Achse des Balls
- 2: bewegt sich auf Y-Achse des Balls

Spieler
wenn Ball in Wahrnehmungsradius, dahin bewegen
Klickbar, wenn geklickt, customizable
radius: 18px

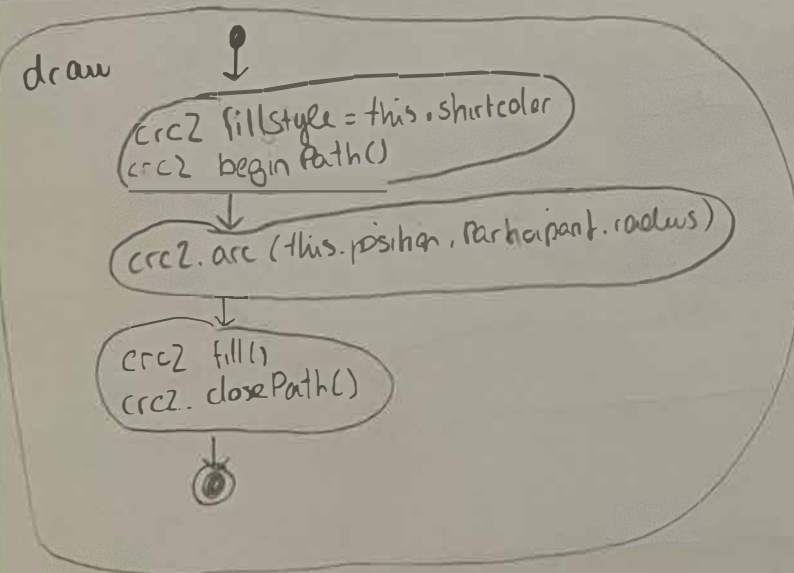
Klassendiagramme Fußballsimulation



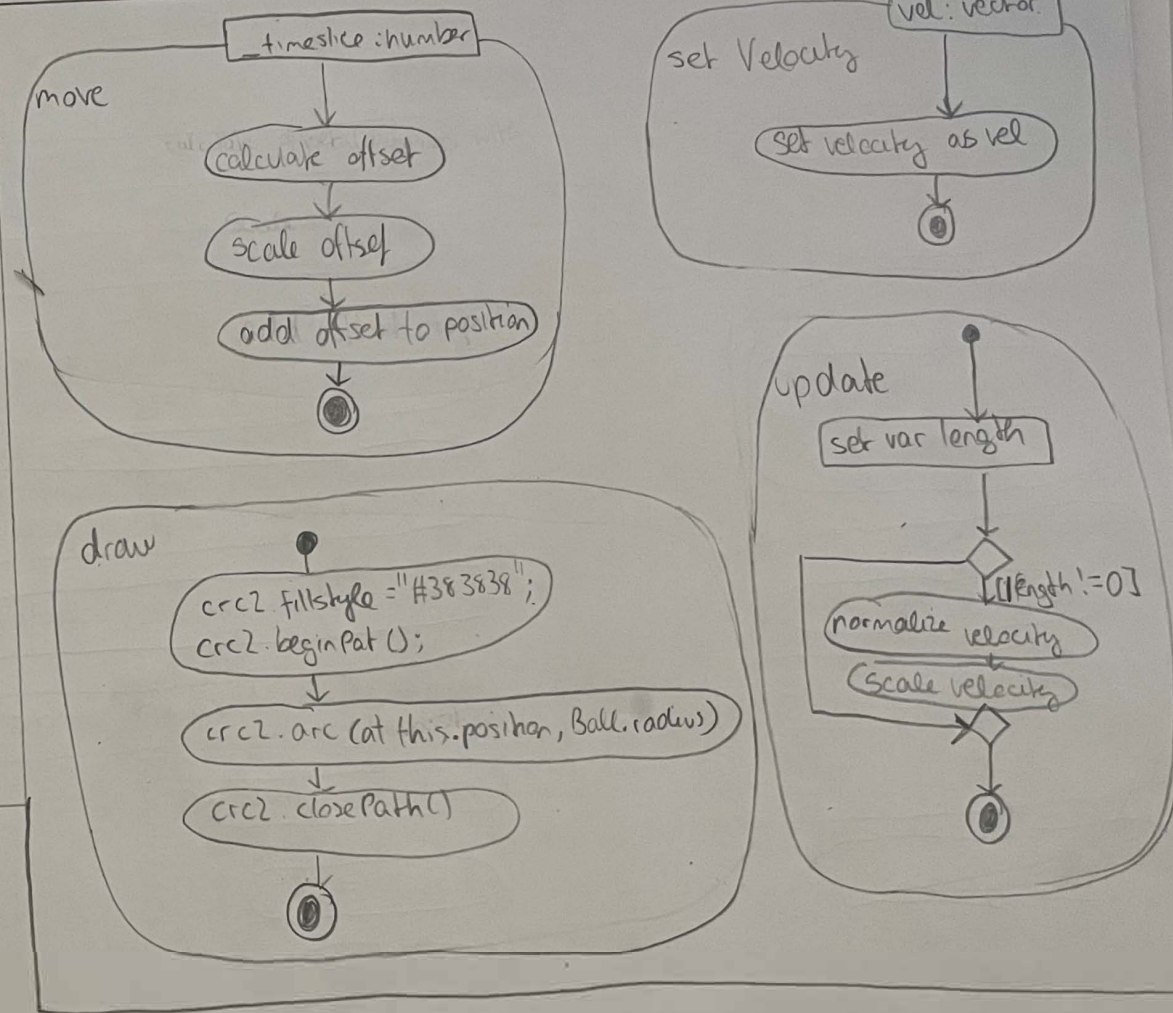
Aktivitätsdiagramm - Moveable



Aktivitätsdiagramm Participant



Aktivitätsdiagramm Ball

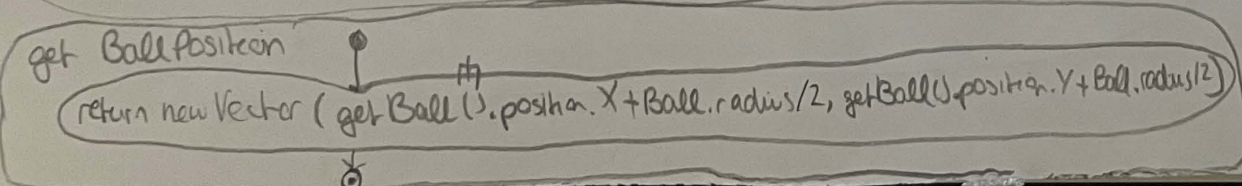
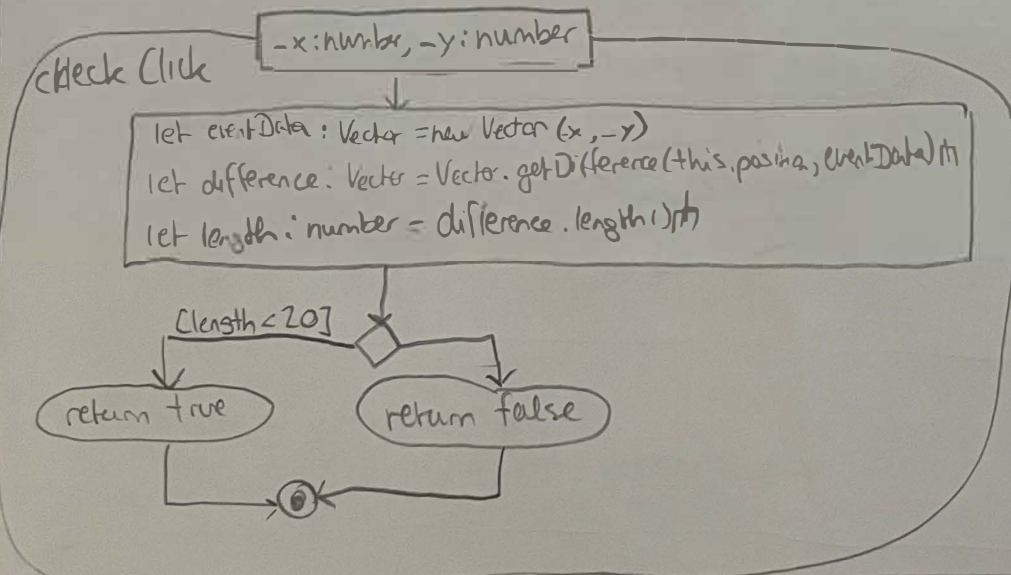
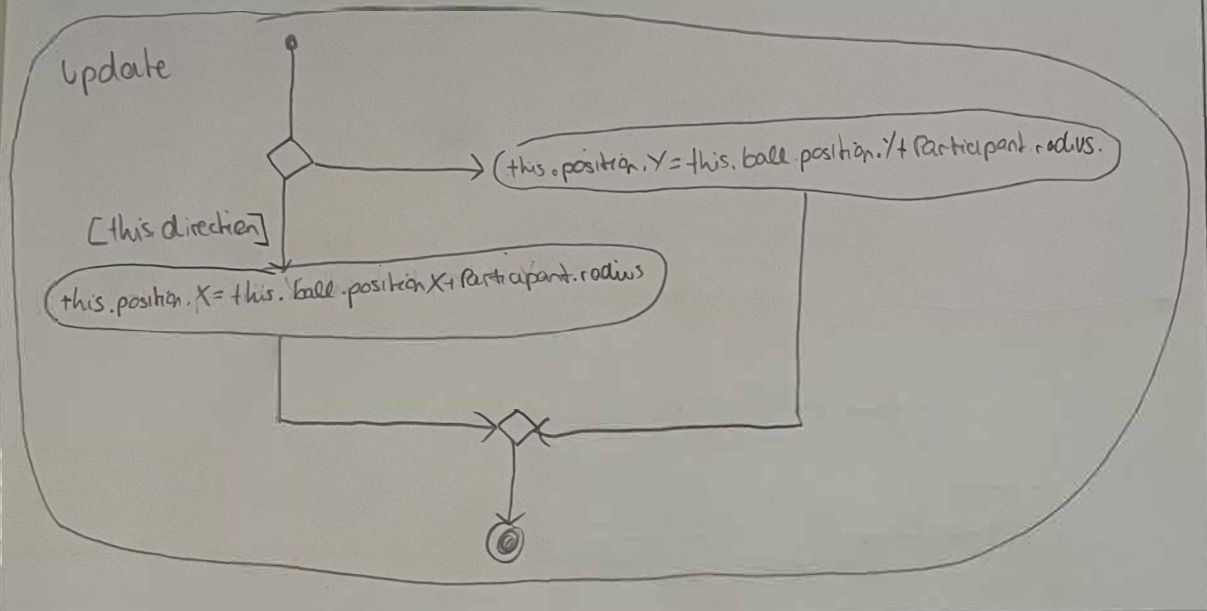


move() → see Aktivitätsdiagramm Ball

update() → see Aktivitätsdiagramm Script.ts

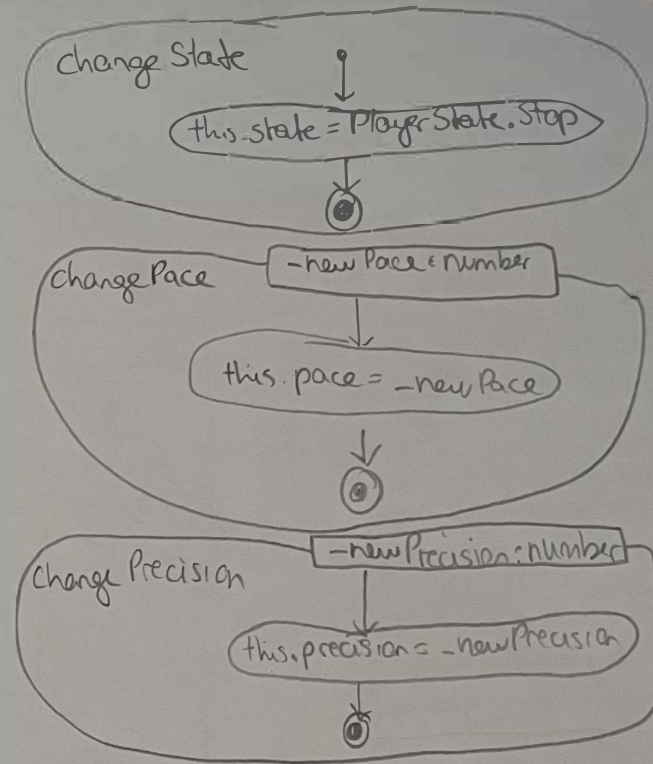
Aktivitätsdiagramm Referee

draw() → see Aktivitätsdiagramm Script.ts



Aktivitätsdiagramm Player

draw() → see Scripts

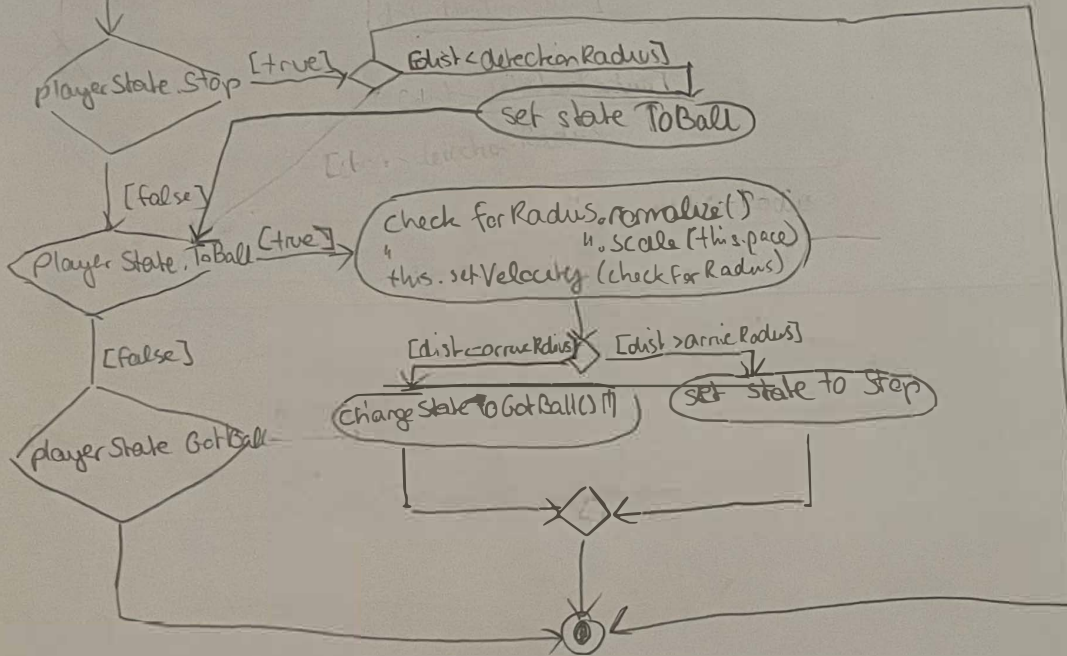


Aktivitätsdiagramme Player (fortgesetzt)

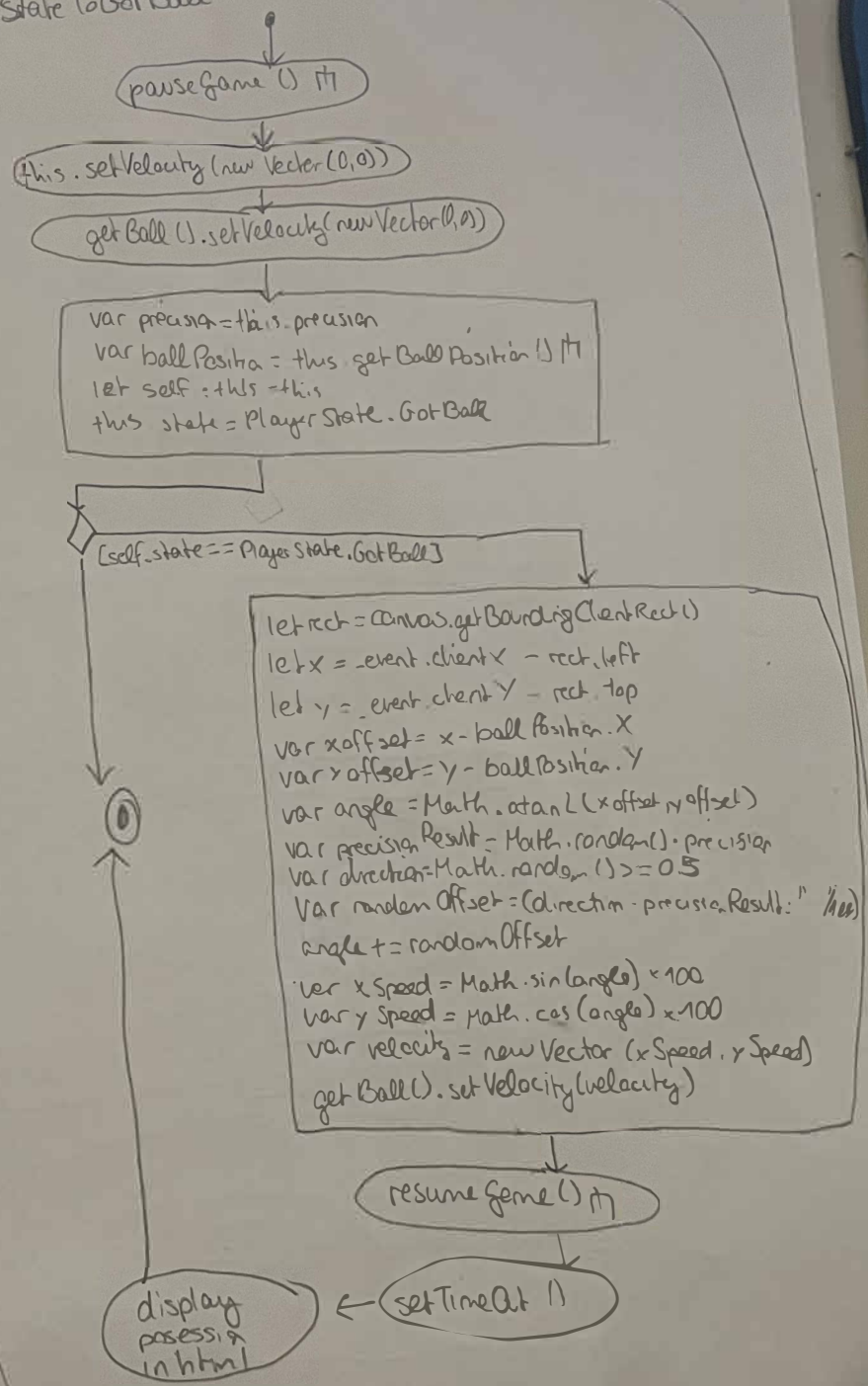
update

```

let checkForRadius: Vector = Vector.getDifference(this.getBallPosition(), this.position)
const dist: number = Math.abs(checkForRadius.length())
const detectionRadius: number = 150
const arriveRadius: number = 10
    
```



changeState To Got Ball



Aktivitätsdiagramm create Players.ts

create Participants

```
playerArray: Participant[]  
ball: Ball  
width: number  
height: number  
color Team 1: string  
color Team 2: string  
min Speed: number  
max Speed: number  
min Precision: number  
max Precision: number
```

```
let playerVelocity: Vector = new Vector(0,0)
```

interface PlayerSkills {

```
name: string  
shirt: string  
number: number  
speed: number = getRandomInt(minSpeed, maxSpeed)  
playerPrecision: number = getRandomInt(minPrecision, maxPrecision)  
team: number }
```

fill array with 22 players

```
let startPositionArrayTeam 1: Vector[]
```

calculate player positions

```
let i = 0
```

```
(i >= startPositionArrayTeam 1.length) ? [i < startPositionArrayTeam 1.length]
```

get Random Int

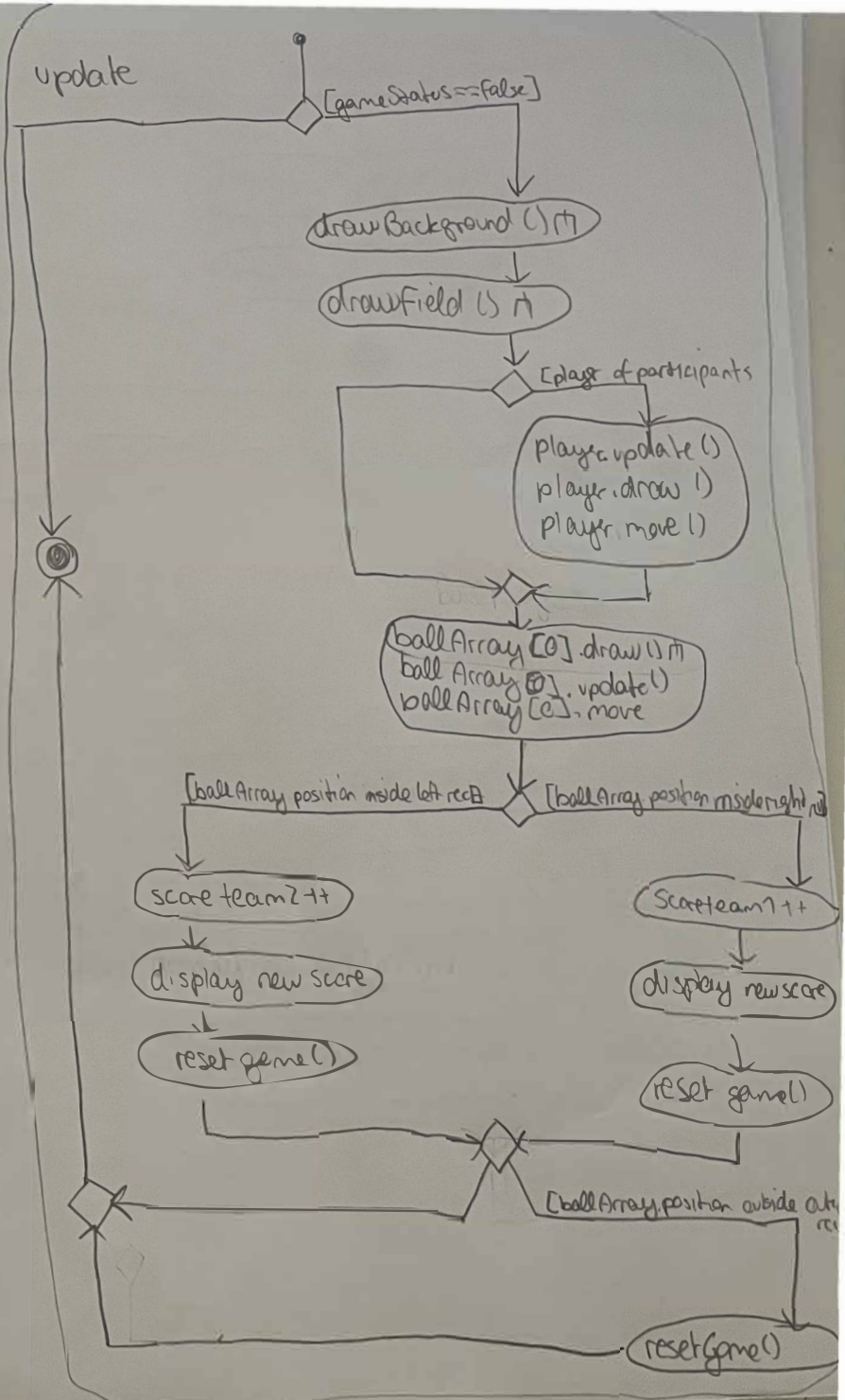
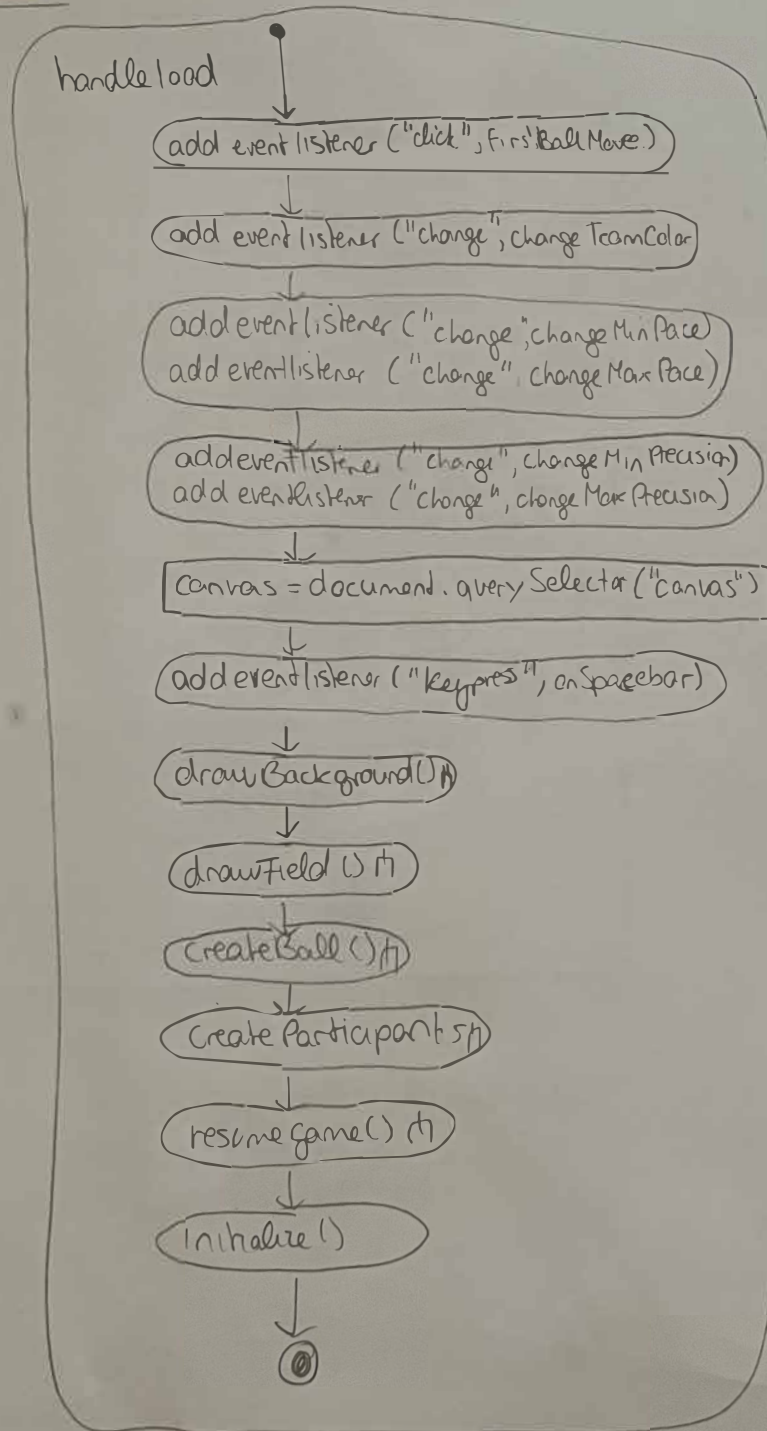
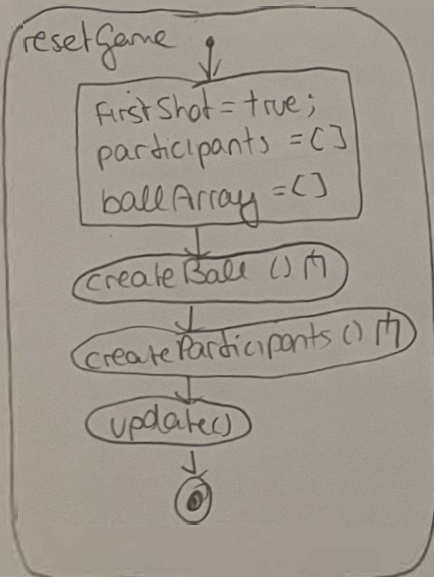
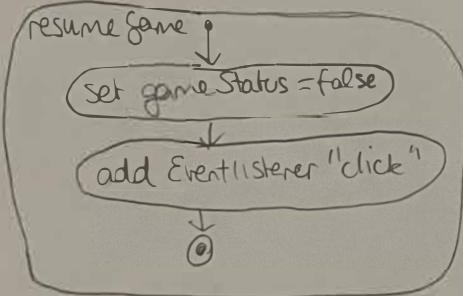
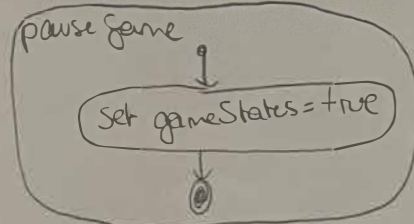
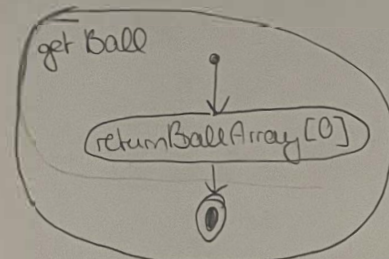
```
min = Math.ceil(min)  
max = Math.floor(max)
```

```
return Math.floor(Math.random() * (max - min + 1) + min)
```

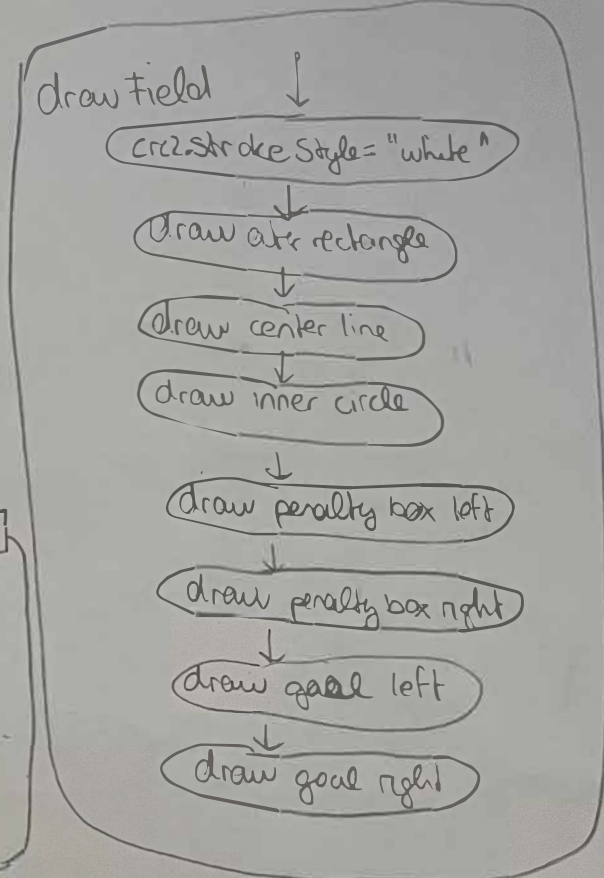
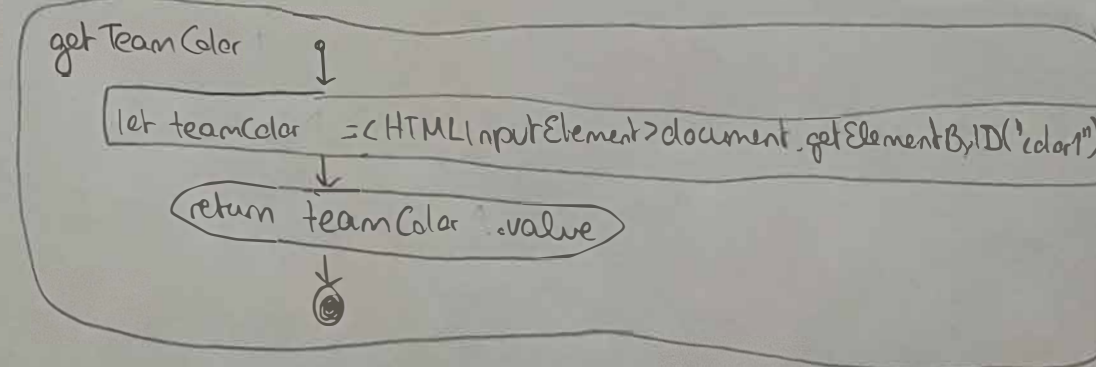
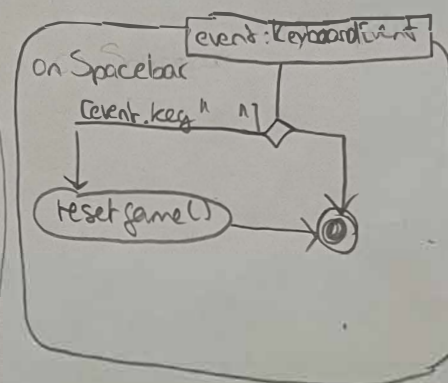
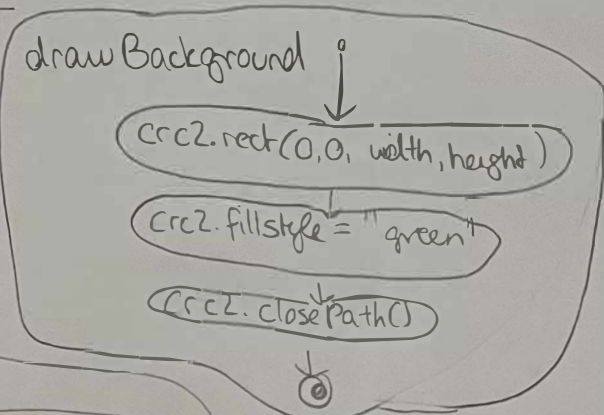
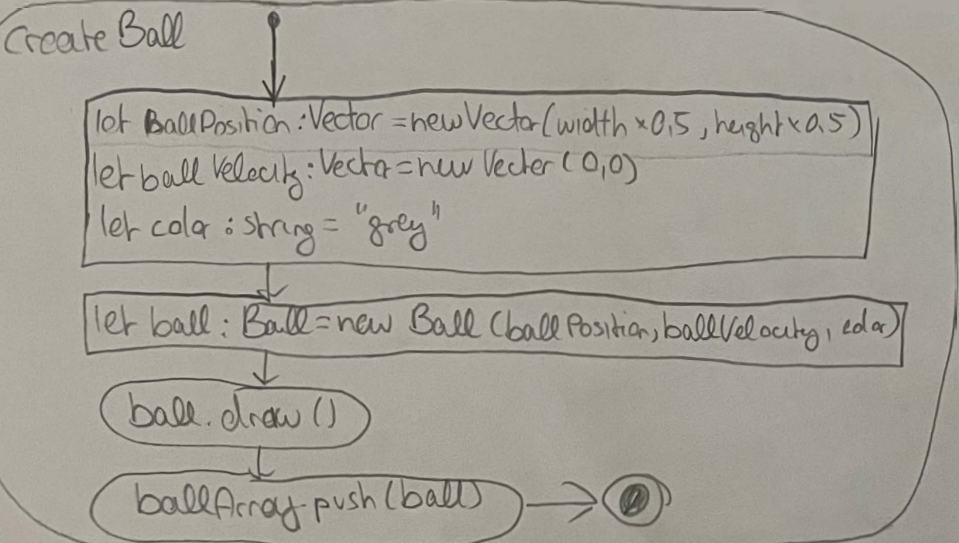
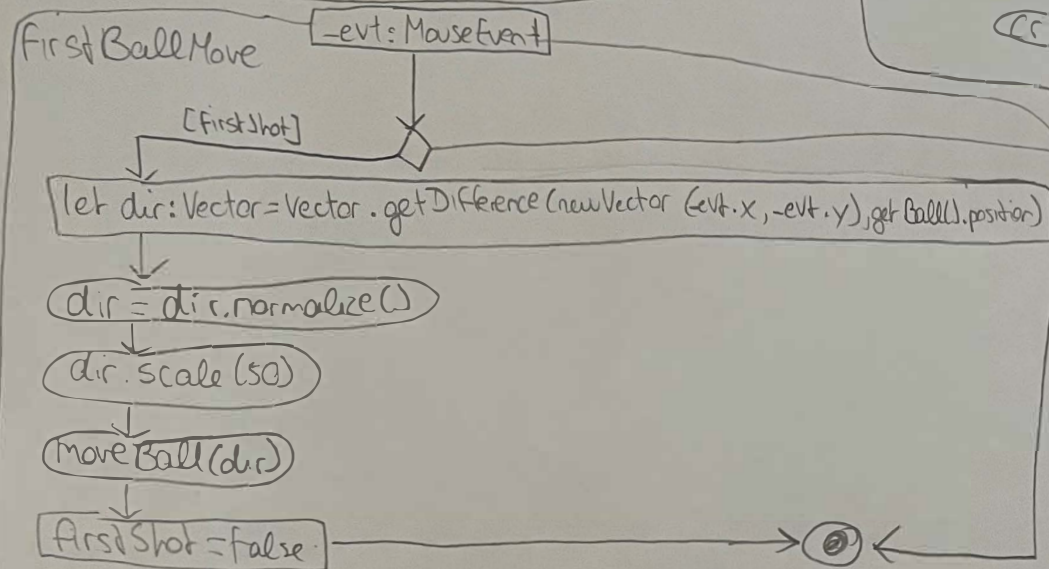
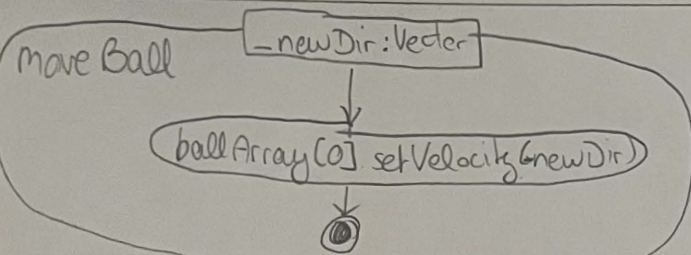
```
let positionReferee = new Vector(425 + participant.radius, 15 + participant.radius)
```

```
playerArray.push(Referee)  
let Referee = new Referee(positionReferee, playerVelocity)
```


Aktivitätsdiagramm Script.ts



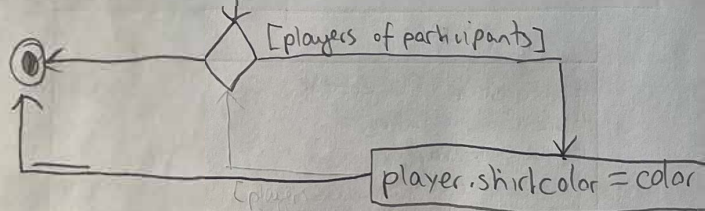
Aktivitätsdiagramm Script.ts (fortgesetzt)



Aktivitätsdiagramm Script.ts (fortgesetzt 2)

Change Team Color

```
let teamColor: HTMLInputElement = document.getElementById("color")  
var color = teamColor.value
```



Change Max Precision

```
let element = getElementById("PrecisionSliderMax")
```

```
maxPrecision = Number(element.value)
```

```
resetGame()
```



Change Min Precision

```
let element = getElementById("PrecisionSliderMin")
```

```
minPrecision = Math.min(Number(element.value), maxPrecision)
```

```
element.value = minPrecision + " "
```

```
resetGame()
```



note: same functions,
different variables to adjust Page

Aktivitätsdiagramm Vector

