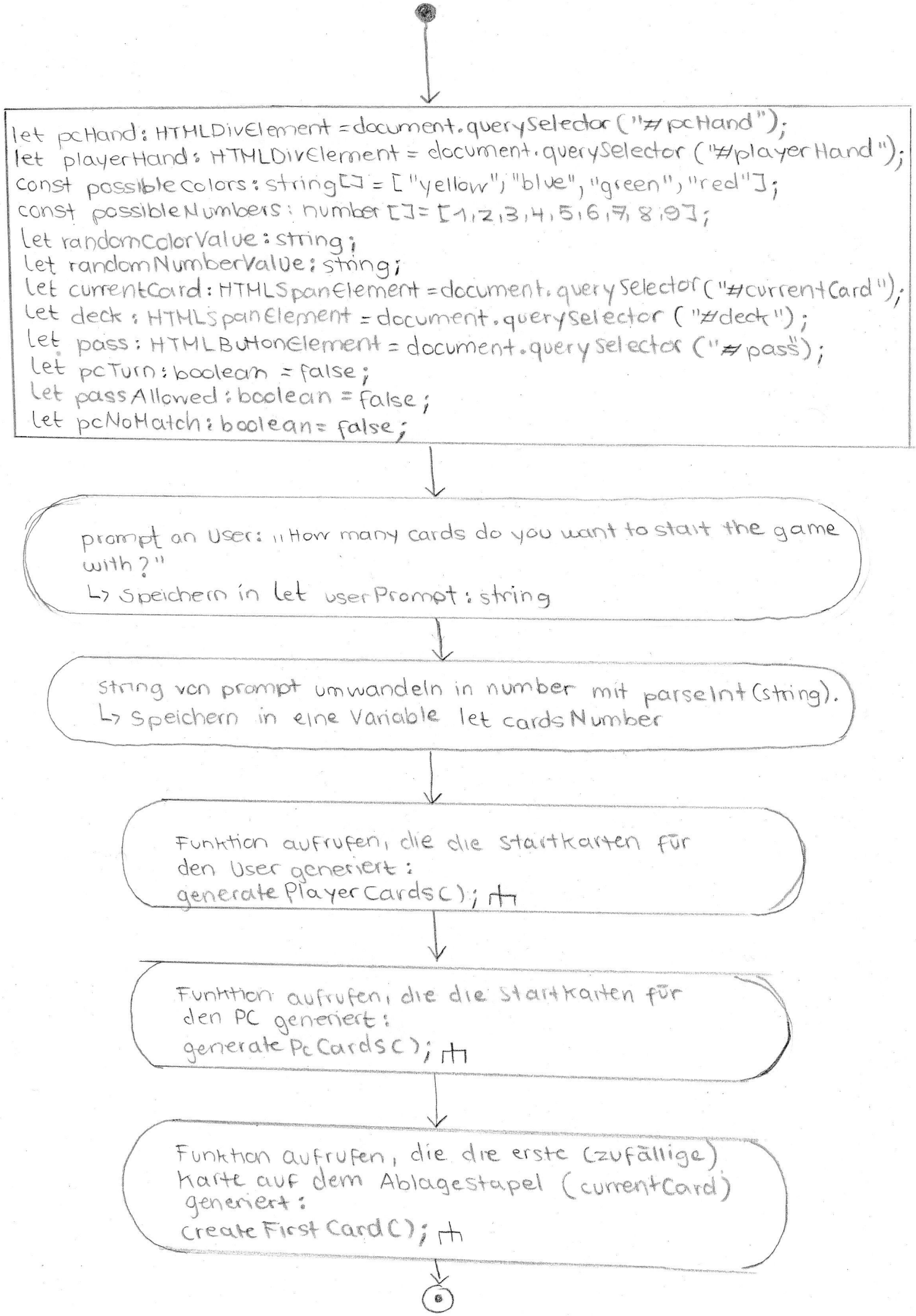
Activity Diagram - Mini Uno



generatePlayerCards

```
function generatePlayerCards():void {
    let i:number=0;
    while (i<cardNumber){
            createRandomValues();
            let card:HTMLSpanElement=document.createElement("span");
            playerHand.appendChild(card);
            card.textContent=randomNumberValue;
            card.className=randomColorValue;
            card.addEventListener("click",chooseCard);
            i++;
            }
}
```

generatePcCards

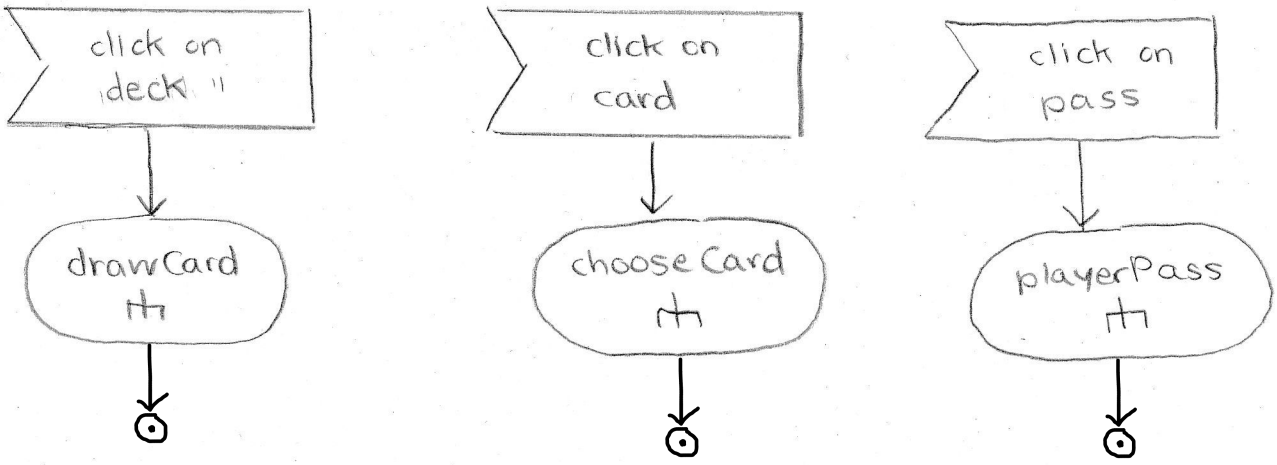
```
function generatePcCards():void {
    let i:number=0;
    while (i<cardNumber){
            createRandomValues();
            let card:HTMLSpanElement=document.createElement("span");
            pcHand.appendChild(card);
            card.textContent=randomNumberValue;
            card.className=randomColorValue;
            card.style.color="lightslategrey";
            card.style.backgroundColor="lightslategrey";
            i++
            }
```

createFirstCard

```
function createFirstCard():void {
    createRandomValues();
    currentCard.textContent=randomNumberValue;
    currentCard.className=randomColorValue;
```

handlePcTurn

```
function handlePcTurn():void {
    if(pcTurn){
        pcMove(); ++
        }
}
```




```
drawCard  -event: MouseEvent

let card: HTMLSpanElement = document.createElement("span");
playerHand.appendChild(card);
createRandomValues();
card.textContent = "randomNumberValue";
card.className = "randomColorValue";
passAllowed = true
```

```
playerPass  -event: MouseEvent

if (passAllowed == true) {
    pcTurn = true;
    handlePcTurn
```

```
chooseCard  -event: MouseEvent

let chosenCard: HTMLSpanElement = -event.target;
if (chosenCard.textContent == currentCard.textContent ||
    chosenCard.className == currentCard.className) {
    currentCard.textContent = chosenCard.textContent;
    currentCard.className = chosenCard.className;
    playerHand.removeChild(chosenCard);
    evaluateWinner();
    pcTurn = true;
    handlePcTurn();
    }
```

CreateRandomValues

```
function createRandomValues():void {
    randomColorValue = possibleColors[Math.floor(Math.random()
                       *possibleColors.length)];
    let x:number = possibleNumbers[Math.floor(Math.random()
                   *possibleNumbers.length)];
    randomNumberValue = x.toString();
```

pcMove

```
function pcMove():void{
    let allCardsPc:HTMLCollection = pcHand.children;
    pcNoMatch = true;

    for(let i:number=0; i<allCardsPc.length ;i++) {
        if(allCardsPc[i].textContent == currentCard.textContent ||
           allCardsPc[i].className == currentCard.className){
            currentCard.textContent = allCardsPc[i].textContent;
            currentCard.className = allCardsPc[i].className;
            pcHand.removeChild(allCardsPc[i]);
            passAllowed = false;
            pcTurn = false;
            pcNoMatch = false;
            break;
        }
    }
    if(pcNoMatch) {
        pcDraw();
        }
```

evaluateWinner(); ⊣

pcDraw

```
function pcDraw():void {
    createRandomValues(); ⊣
    let card:HTMLSpanElement = document.createElement("span");
    pcHand.appendChild(card);
    card.textContent = randomNumberValue;
    card.className = randomColorValue;
    card.style.color = "lightslategrey";
    card.style.backgroundColor = "lightslategrey";

    if(card.textContent == currentCard.textContent ||
       card.className == currentCard.className) {
        currentCard.textContent = card.textContent;
        currentCard.className = card.className;
        pcHand.removeChild(card);
    }
    pcTurn = false
    passAllowed = false
```

evaluateWinner

```
function evaluateWinner(): void {
    let allCardsPc: HTMLCollection = pcHand.children;
    let allCardsPlayer: HTMLCollection = playerHand.children;

    if(allCardsPc.length == 0) {
        alert("Du hast verloren :( ");
        location.reload();
    }

    if(allCardsPlayer.length == 0) {
        alert("Du hast gewonnen");
        location.reload();
    }
}
```