

Министерство образования Республики Беларусь  
Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Дисциплина: Современные языки программирования

ЛАБОРАТОРНАЯ РАБОТА № 8

Выполнил:

Атаев И.М. гр. 910101

Проверила:

Василькова А.Н.

Минск 2022

Задание: Браузерные события. Всплытие и погружение. Делегирование событий. Генерация событий. События мыши. Мышь: mouseover/out, mouseenter/leave. Drag'n'Drop с событиями мыши. Прокрутка: событие scroll. Клавиатура: keyup, keydown.

Задание:

Реализуйте интерфейс игры «Дартс», в соответствии с примером.

1. Мяч появляется на красной линии случайным образом
2. Игрок перетаскивает мяч мышкой и отпускает, целясь в мишень.
3. Область перетаскивания мяча мышкой должна быть ограничена: нельзя перетаскивать мяч за красную черту.
4. Разработайте функцию `animate()` для анимации полета мяча. Вначале мяч летит с ускорением, затем замедляется, приближаясь в мишени.
5. Если мяч попадает в мишень, выводится сообщение (Попал!)
6. Если мяч не попадает в мишень, мяч возвращается в исходную позицию на красную линию

Листинг кода:

```
const ballPosition = Math.random() * (90 - 10) + 10;

const ballNode = document.querySelector(".ball");
const field = document.querySelector(".field");
const board = document.querySelector(".board");

const fieldHeight = document
  .querySelector(".field")
  .getBoundingClientRect().height;

const zoneHeight = document
  .querySelector(".zone")
  .getBoundingClientRect().height;

ballNode.style.left = `${ballPosition}%`;

const ballCenter = [
  (ballNode.getBoundingClientRect().right +
    ballNode.getBoundingClientRect().x) /
    2,
  (ballNode.getBoundingClientRect().bottom +
    ballNode.getBoundingClientRect().y) /
    2,
];

function endpoints() {
  const diffX = ballCenter[0] - ballNode.getBoundingClientRect().left - 35;
  const diffY = ballNode.getBoundingClientRect().top - ballCenter[1];

  const coordinateX = 4 * diffX + ballCenter[0];
  const coordinateY = 4 * diffY;

  ballNode.style.transition = 0.2 + "s";
```

```

ballNode.style.left = coordinateX + "px";
ballNode.style.top = coordinateY + "px";

if (
  board.getBoundingClientRect().left < coordinateX &&
  coordinateX < board.getBoundingClientRect().right &&
  board.getBoundingClientRect().top < coordinateY &&
  coordinateY < board.getBoundingClientRect().bottom
)
  setTimeout(() => alert("Вы попали в мишень!"), 200);
}

ballNode.onmousedown = function (event) {
  let shiftX = event.clientX - ballNode.getBoundingClientRect().left;
  let shiftY = event.clientY - ballNode.getBoundingClientRect().top;

  let distance = null;

  document.body.append(ballNode);

  moveAt(event.pageX, event.pageY);

  function moveAt(pageX, pageY) {
    distance = Math.sqrt(
      Math.pow(pageX - ballCenter[0], 2) + Math.pow(pageY - ballCenter[1], 2)
    );
  }

  if (
    fieldHeight > pageY - shiftY ||
    zoneHeight + fieldHeight - ballNode.width / 2 < pageY - shiftY ||
    pageX - shiftX > window.innerWidth - ballNode.width ||
    distance > 150
  )
    return null;

  ballNode.style.left = pageX - shiftX + "px";
  ballNode.style.top = pageY - shiftY + "px";
}

function onMouseMove(event) {
  moveAt(event.pageX, event.pageY);
}

document.addEventListener("mousemove", onMouseMove);

ballNode.onmouseup = function () {
  endpoints(distance);

  document.removeEventListener("mousemove", onMouseMove);

  ballNode.onmouseup = null;
};

ballNode.ondragstart = function () {
  return false;
};

```

## Результаты работы программы:

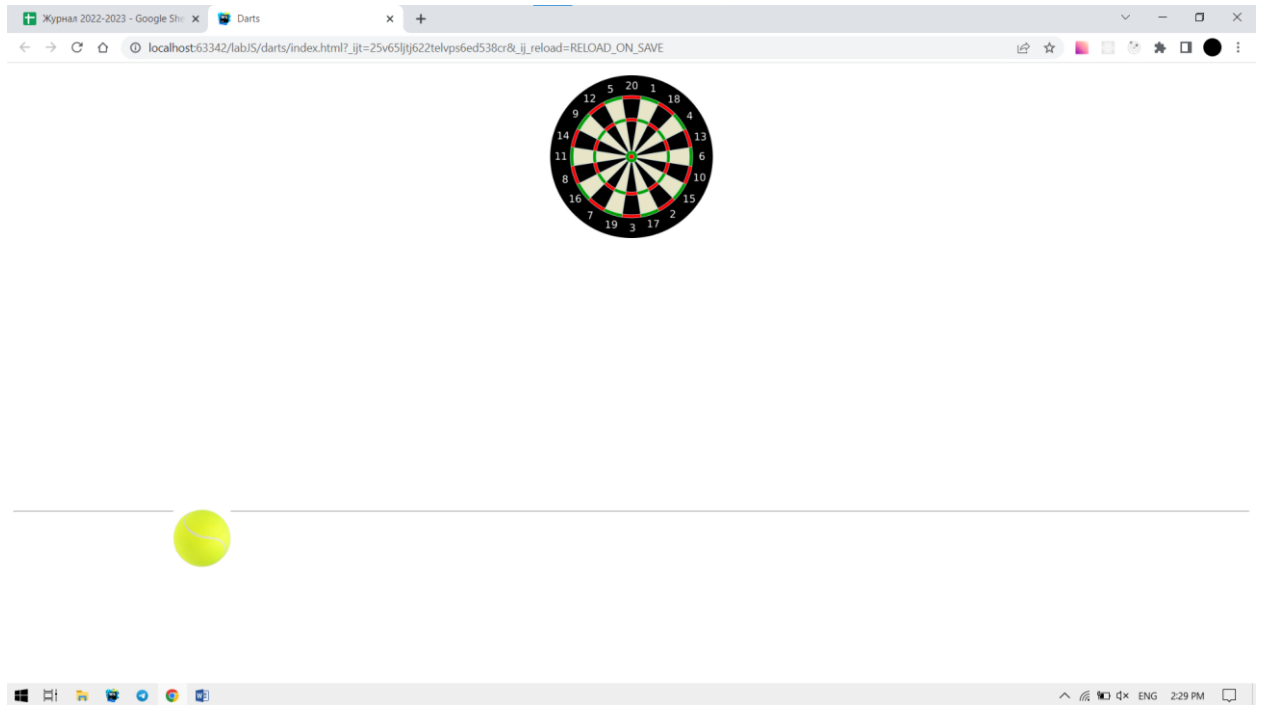


Рис. 1 – Drag'n'Drop.

Вывод: В ходе выполнения лабораторной работы ознакомились с Браузерные события. Всплытие и погружение. Делегирование событий. Генерация событий. События мыши. Мышь: mouseover/out, mouseenter/leave. Drag'n'Drop с событиями мыши. Прокрутка: событие scroll. Клавиатура: keyup, keydown.