Gmail Mayank Raj <mayankrajnit.0405@gmail.com>

Droid Defense

1 message

Mayank Raj <21mayankraj12d@gmail.com> Mon, Jun 12, 2023 at 11:19 PM
To: Mayank Raj <mayankrajnit.0405@gmail.com>

```
<!DOCTYPE html>
<html>
<head>
 <title>Bot Attack Game</title>
 <body>
 <div class="body">
 <div><h1>Droid Defense</h1></div>
 <div id="board"></div>
 </div>
</body>
 <style>
 #game-container {
 width: 590px;
 height: 590px;
 border: 1px solid #000;
 position: relative;
 background: linear-gradient(purple,rgb(120, 90, 90));
 }


 #player {
 background-color: purple;
 border: .25vmin solid white;
 border-radius: 12px;
 width: 20px;
 height: 20px;
 position: absolute;
 position: absolute;
 bottom: 0;
 left: 50%;
 transform: translateX(-50%);
 width: 40px;
 height: 40px;
 background-color: blue;
 }

 .bot {
 position: absolute;
 width: 30px;
 height: 30px;
```

```
    border-radius: 12px;
    background-color: red;
    }

    #base {
    position: absolute;
    bottom: 10px;
    left: 50%;
    transform: translateX(-50%);
    width: 80px;
    height: 50px;
    background-color: yellow;
    }

    #score {
    margin-top: 10px;
    text-align: center;
    }
    </style>
</head>
<body>
    <div id="game-container">
    <div id="player"></div>
    <div id="base"></div>
    </div>
    <div id="score">Score: 0</div>

    <script>
    // Game variables
    var player;
    var base;
    var bots = [];
    var score = 0;
    var bullets=[]


    // Game configuration
    var gameContainer = document.getElementById("game-container");
    var playerSpeed = 5;
    var botSpeed = 1;
    var botSpawnInterval = 1000; // milliseconds
    var playerHealth = 3;
    var baseHealth = 10;

    // Player controls
    var leftPressed = false;
    var rightPressed = false;

    // Create player
    function createPlayer() {
```

```
player = document.getElementById("player");
}

// Move player
function movePlayer() {
if (leftPressed) {
var newPos = player.offsetLeft - playerSpeed;
if (newPos >= 0) {
player.style.left = newPos + "px";
}
}
if (rightPressed) {
var newPos = player.offsetLeft + playerSpeed;
if (newPos <= gameContainer.offsetWidth - player.offsetWidth) {
player.style.left = newPos + "px";
}
}
}

// Shoot
function shoot() {
var bullet = document.createElement("div");
bullet.classList.add("bullet");
bullet.style.left = player.offsetLeft + player.offsetWidth / 2 + "px";
bullet.style.top = player.offsetTop - 10 + "px";
gameContainer.appendChild(bullet);

var bulletInterval = setInterval(function() {
var bullets = document.getElementsByClassName("bullet");
for (var i = 0; i < bullets.length; i++) {
var bullet = bullets[i];
var bulletPos = bullet.offsetTop;

// Check collision with bots
for (var j = 0; j < bots.length; j++) {
var bot = bots[j];
var botPos = bot.offsetTop;

if (bulletPos <= botPos + bot.offsetHeight && bulletPos >= botPos) {
gameContainer.removeChild(bullet);
gameContainer.removeChild(bot);
bots.splice(j, 1);
score += 10;
document.getElementById("score").innerText = "Score: " + score;
break;
}
}

bullet.style.top = bulletPos - 10 + "px";
```

```
// Check if bullet is out of the game container
if (bulletPos <= 0) {
gameContainer.removeChild(bullet);
clearInterval(bulletInterval);
}
}
}, 10);
}

// Create bot
function createBot() {
var bot = document.createElement("div");
bot.classList.add("bot");
bot.style.left = Math.floor(Math.random() * (gameContainer.offsetWidth - 30)) + "px";
bot.style.top = "0px";
gameContainer.appendChild(bot);
bots.push(bot);
}

// Move bots
function moveBots() {
for (var i = 0; i < bots.length; i++) {
var bot = bots[i];
var botPos = bot.offsetTop + botSpeed;
bot.style.top = botPos + "px";

// Check collision with player
if (
botPos + bot.offsetHeight >= player.offsetTop &&
botPos <= player.offsetTop + player.offsetHeight &&
bot.offsetLeft + bot.offsetWidth >= player.offsetLeft &&
bot.offsetLeft <= player.offsetLeft + player.offsetWidth
) {
gameContainer.removeChild(bot);
bots.splice(i, 1);
playerHealth--;
if (playerHealth <= 0) {
endGame();
}
break;
}

// Check collision with base
if (
botPos + bot.offsetHeight >= base.offsetTop &&
botPos <= base.offsetTop + base.offsetHeight &&
bot.offsetLeft + bot.offsetWidth >= base.offsetLeft &&
bot.offsetLeft <= base.offsetLeft + base.offsetWidth
) {
gameContainer.removeChild(bot);
```

```
bots.splice(i, 1);
baseHealth--;
if (baseHealth <= 0) {
endGame();
}
break;
}

// Check if bot is out of the game container
if (botPos >= gameContainer.offsetHeight) {
gameContainer.removeChild(bot);
bots.splice(i, 1);
break;
}
}
}

// Game loop
function gameLoop() {
movePlayer();
moveBots();
}

// Game start
function startGame() {
createPlayer();
base = document.getElementById("base");

// Keyboard controls
document.addEventListener("keydown", function(event) {
if (event.code === "ArrowLeft") {
leftPressed = true;
}
if (event.code === "ArrowRight") {
rightPressed = true;
}
if (event.code === "Enter") {
shoot();
}
});

document.addEventListener("keyup", function(event) {
if (event.code === "ArrowLeft") {
leftPressed = false;
}
if (event.code === "ArrowRight") {
rightPressed = false;
}
});
```

```
// Bot spawning
setInterval(function() {
createBot();
}, botSpawnInterval);

// Game loop
setInterval(function() {
gameLoop();
}, 10);
}

// End game
function endGame() {
// Clear game interval
clearInterval();

// Show game over message
alert("Game Over!\nFinal Score: " + score);

// Reset game state
playerHealth = 3;
baseHealth = 10;
score = 0;
document.getElementById("score").innerText = "Score: " + score;

// Remove remaining bots and bullets
var remainingBots = document.getElementsByClassName("bot");
while (remainingBots.length > 0) {
gameContainer.removeChild(remainingBots[0]);
}

var remainingBullets = document.getElementsByClassName("bullet");
while (remainingBullets.length > 0) {
gameContainer.removeChild(remainingBullets[0]);
}

// Restart game
startGame();
}

// Start the game
startGame();
</script>
</body>
</html>
```