File: index.html

```
<html>
    <script type="text/javascript" src="gsap/TweenMax.js"> </script>
    <script type="text/javascript" src="js/game.js"> </script>
    <style>
        .gamification-label {
            position:absolute;
            font-size:30px;
            width:25%;
            height:25%;
            display:block;
            text-align: center;
            background-color:blue;
            color: white;
        }
        .gamification-meter {
            position:absolute;
            font-size:30px;
            width:25%;
            height:75%;
            display:block;
            top:5%;
            text-align: center;
            background-color:black;
            border-style:solid;
            color: yellow;
            word-wrap: break-word;
        }
        #global-container {
            border-style:double;
            width:90%;
            height:90%;
            margin:0 auto;
        }
        #gamification-meters-container {
            border-style:dashed;
            width:90%;
            height:20%;
            margin:10px auto;
            position:relative;
        }
        #game-container {
            border-style:double;
            width:80%;
            height:70%;
            margin:10px auto;
            position:relative;
            visibility: hidden;
        }
        Button {
            width: 200px;
```

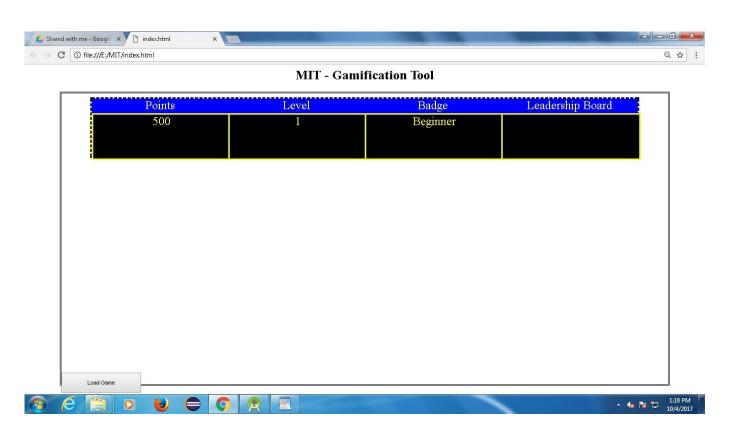
```
height: 50px;
          left: 50%;
       }
   </style>
   <h1 align="center"> MIT - Gamification Tool </h1>
   <div id="global-container"">
       <div id="gamification-meters-container">
          <div id= "labels">
              <label id="lblpoints" class="gamification-label"</pre>
style="left:0%"> Points </label>
              <label id="lbllevel" class="gamification-label"</pre>
style="left:25%"> Level </label>
              <label id="lblbadge" class="gamification-label"</pre>
style="left:50%"> Badge </label>
              <label id="lblleadership" class="gamification-label"</pre>
style="left:75%"> Leadership Board </label>
          </div>
          <div id= "meters">
              style="left:0%"> xx </label>
              style="left:25%"> xx </label>
              style="left:50%"> xx </label>
              style="left:75%"> xx </label>
          </div>
       </div>
       <div id="game-container">
          <svg height="100%" width="100%">
              <circle id="game-ball" cx="25" cy="25" r="25"</pre>
stroke="black" stroke-width="1" fill="red"/>
              <rect id="points-brick" width="50" height="50"</pre>
style="fill:gold;stroke-width:3;stroke:rgb(0,0,0)"/>
              <rect id="platform" width="600" height="20"</pre>
style="fill:rqb(0,0,255);stroke-width:3;stroke:rqb(0,0,0)"/>
          </sva>
       </div>
       <Button id="load-game-button" name="Load Game" type="submit"> Load
Game </Button>
   </div>
</html>
```

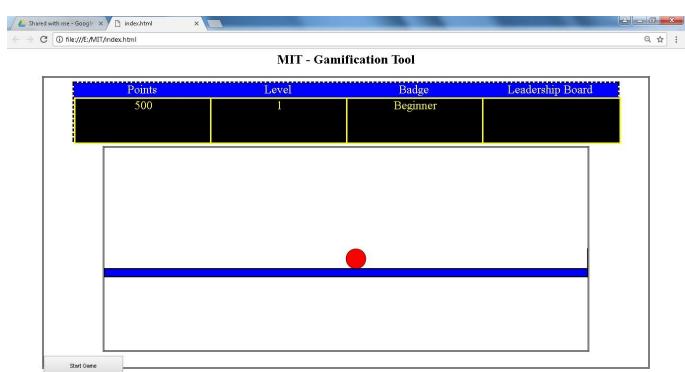
File: game.js

```
var loadGameButton;
var gameContainerDiv;
var gameBall;
var platform;
var gameBallHeight;
var pointsBrick;
var meterPoints;
var meterLevel;
var meterBadge;
var meterLeadership;
var onLoad = this.onLoad;
var onLoadGame = this.onLoadGame;
var onStartGame = this.onStartGame;
var updateMeters = this.updateMeters;
var handleMouseClick = this.handleMouseClick;
var points = 500;
var level = 1;
var badge = "Beginner";
var leadership;
var tweenBall;
var tweenBrick;
window.onload = onLoad;
function onLoad() {
    loadGameButton = document.getElementById("load-game-button");
    qameContainerDiv = document.getElementById("game-container");
    gameBall = document.getElementById("game-ball");
    platform = document.getElementById("platform");
    pointsBrick = document.getElementById("points-brick");
    meterPoints = document.getElementById("meterpoints");
    meterLevel = document.getElementById("meterlevel");
    meterBadge= document.getElementById("meterbadge");
    meterLeadership = document.getElementById("meterleadership");
    updateMeters();
    loadGameButton.onclick = onLoadGame;
    gameBallHeight = gameBall.r.baseVal.value * 2;
    TweenMax.set(gameBall, {x: gameContainerDiv.clientWidth/2 + 'px', y:
gameContainerDiv.clientHeight/2 + 'px'});
    TweenMax.set(platform, {width: gameContainerDiv.clientWidth + 'px', y:
gameContainerDiv.clientHeight/2 + gameBallHeight + 'px'});
    TweenMax.set(pointsBrick, {x: gameContainerDiv.clientWidth + 'px', y:
gameContainerDiv.clientHeight/2 + 'px'});
```

```
}
function addMouseEvents(){
    gameContainerDiv.addEventListener("click", mouseclick)
}
function updateMeters(){
    meterpoints.innerHTML = points.toString();
    meterLevel.innerHTML = level.toString();
    meterBadge.innerHTML = badge;
    meterLeadership.innerHTML = "";
}
function onLoadGame() {
    gameContainerDiv.style.visibility = "visible";
    loadGameButton.innerHTML = "Start Game";
    loadGameButton.onclick = onStartGame;
}
function onStartGame() {
    loadGameButton.innerHTML = "Bounce";
    loadGameButton.onclick = handleMouseClick;
    tweenBrick = TweenMax.to(pointsBrick, 2, {x: 'Opx', yoyo: true,
repeat: -1, onUpdate : () => {
            if (Math.abs(pointsBrick. gsTransform.x -
gameBall. gsTransform.x) < 50</pre>
            && Math.abs(pointsBrick.gsTransform.y -
gameBall._gsTransform.y) < 50) {</pre>
                tweenBrick.kill();
                if(tweenBall)
                    tweenBall.kill();
        }
     });
}
function handleMouseClick() {
    if(tweenBall)
        tweenBall.kill();
    tweenBall = TweenMax.to(gameBall, 0.20, {y:
gameContainerDiv.clientHeight/4 + 'px', yoyo: true, repeat: 1, onComplete:
() => \{
            points = points + 5;
            updateMeters();
        }
    });
}
```

OUTPUT:





▲ 🗽 👺 🖫 1:19 PM

