```
<!doctype html>
<html lang="pt-BR">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0,
user-scalable=no">
<title>Jogo de Tiro 2D - GitHub Pages</title>
<style>
html, body {margin:0; padding:0; height:100%; overflow:hidden; font-family: Arial, Helvetica,
sans-serif; background:#111; color:#eee;}
#menu, #gameWrap {position:absolute; top:0; left:0; width:100%; height:100%; display:flex;
align-items:center; justify-content:center; flex-direction:column;}
canvas {border:0; box-shadow:0 8px 30px rgba(0,0,0,0.6);}
.btn {display:inline-block; padding:14px 24px; margin:10px; background:#1f6feb;
border-radius:8px; cursor:pointer; font-size:20px;}
.btn.secondary {background:#2b2b2b;}
#configPanel {display:none; flex-direction:column; margin-top:20px;}
select {padding:8px; margin-top:8px; border-radius:6px; background:#222; color:#fff;
border:none; font-size:16px;}
</style>
</head>
<body>

<div id="menu">
  <h1>Jogo de Tiro 2D</h1>
  <span class="btn" id="playBtn">Jogar</span>
  <span class="btn secondary" id="configBtn">Configurações</span>
  <div id="configPanel">
    <label>Qualidade da resolução:</label>
    <select id="qualitySelect">
      <option value="low">Baixa</option>
      <option value="medium" selected>Média</option>
      <option value="high">Alta</option>
    </select>
  </div>
</div>

<div id="gameWrap" style="display:none; flex-direction:column; align-items:center;
justify-content:center;">
  <canvas id="canvas"></canvas>
</div>

<script>
const menu = document.getElementById('menu');
const gameWrap = document.getElementById('gameWrap');
const canvas = document.getElementById('canvas');
const ctx = canvas.getContext('2d');
let WIDTH = window.innerWidth;
let HEIGHT = window.innerHeight;
canvas.width = WIDTH; canvas.height = HEIGHT;

let player=null, bullets=[], enemies=[], money=500;
let
spawnPoints=[{x:80,y:80},{x:WIDTH-80,y:HEIGHT-80},{x:80,y:HEIGHT-80},{x:WIDTH-80,y:8
0},{x:WIDTH/2,y:HEIGHT/2}];
let shopSelection='pistol';
const
WEAPONS={pistol:{cost:0,rate:400,spd:8,damage:15,spread:6},rifle:{cost:300,rate:160,spd:1
0,damage:22,spread:3},shotgun:{cost:450,rate:900,spd:7,damage:12,spread:18,pellets:6}};
let keys={}, mouse={x:0,y:0,down:false}, mode='desktop';
let quality='medium';
let buildings=[], mountains=[];

// --- Menu ---
document.getElementById('playBtn').addEventListener('click',()=>{
  menu.style.display='none';
  gameWrap.style.display='flex';
  spawnPlayer();
  generateMap();
});
document.getElementById('configBtn').addEventListener('click',()=>{
  const panel=document.getElementById('configPanel');
  panel.style.display=panel.style.display==='none'?'flex':'none';
});
document.getElementById('qualitySelect').addEventListener('change',e=>{quality=e.target.va
lue;});

// --- Player spawn ---
function spawnPlayer(){
  const p=spawnPoints[Math.floor(Math.random()*spawnPoints.length)];

player={x:p.x,y:p.y,w:18,h:18,hp:100,angle:0,speed:2.4,weapon:shopSelection,lastShot:0,vy:
0,onGround:true,crouch:false};
}

// --- Controls ---
window.addEventListener('keydown',e=>{keys[e.key.toLowerCase()]=true;});
window.addEventListener('keyup',e=>{keys[e.key.toLowerCase()]=false;});
canvas.addEventListener('mousemove',e=>{const r=canvas.getBoundingClientRect();
mouse.x=e.clientX-r.left; mouse.y=e.clientY-r.top;});
canvas.addEventListener('mousedown',e=>{if(e.button===0)mouse.down=true;});
canvas.addEventListener('mouseup',e=>{if(e.button===0)mouse.down=false;});

// --- Shooting ---
function shoot(x,y,a,wname){
  const w=WEAPONS[wname];
  const now=Date.now();
  if(now-player.lastShot<w.rate)return;
  player.lastShot=now;
  if(wname==='shotgun'){
    for(let i=0;i<w.pellets;i++){
      const ang=a+(Math.random()*2-1)*w.spread*Math.PI/180;
      bullets.push({x,y,dx:Math.cos(ang)*w.spd,dy:Math.sin(ang)*w.spd,damage:w.damage});
    }
  } else {
    const ang=a+(Math.random()*2-1)*w.spread*Math.PI/180;
    bullets.push({x,y,dx:Math.cos(ang)*w.spd,dy:Math.sin(ang)*w.spd,damage:w.damage});
  }
}

// --- Map generation (realista simples) ---
function generateMap(){
  buildings=[]; mountains=[];
  for(let i=0;i<20;i++){
    buildings.push({x:Math.random()*(WIDTH-100)+50, y:Math.random()*(HEIGHT-100)+50,
w:40+Math.random()*40, h:40+Math.random()*40, color:'#888'});
  }
  for(let i=0;i<5;i++){
    mountains.push({x:Math.random()*(WIDTH-100)+50, y:Math.random()*(HEIGHT-100)+50,
r:40+Math.random()*30, color:'#555'});
  }
}

// --- Game loop ---
function update(){
  if(player){
    let mvx=0,mvy=0;
    if(keys['w'])mvy-=1; if(keys['s'])mvy+=1;
    if(keys['a'])mvx-=1; if(keys['d'])mvx+=1;
    player.angle=Math.atan2(mouse.y-player.y,mouse.x-player.x);
    if(mouse.down) shoot(player.x,player.y,player.angle,player.weapon);
    player.speed = keys['control'] ? 4 : 2.4;
    player.crouch = keys['shift'];
    if(keys[' ']){if(player.onGround){player.vy=-6;player.onGround=false;}}
    player.vy+=0.25; player.y+=player.vy;
    if(player.y>HEIGHT-20){player.y=HEIGHT-20; player.vy=0; player.onGround=true;}
    const mag=Math.hypot(mvx,mvy)||1; player.x+=(mvx/mag)*player.speed;
    player.h = player.crouch?12:18;
    player.x=Math.max(10,Math.min(WIDTH-10,player.x));
player.y=Math.max(10,Math.min(HEIGHT-10,player.y));
  }
}

function draw(){
  ctx.clearRect(0,0,WIDTH,HEIGHT);
  ctx.fillStyle='#4caa4c'; ctx.fillRect(0,0,WIDTH,HEIGHT); // chão verde
  mountains.forEach(m=>{
    ctx.beginPath(); ctx.fillStyle=m.color;
    ctx.moveTo(m.x,m.y); ctx.lineTo(m.x-50,m.y+50); ctx.lineTo(m.x+50,m.y+50);
    ctx.closePath(); ctx.fill();
  });
  buildings.forEach(b=>{ctx.fillStyle=b.color; ctx.fillRect(b.x,b.y,b.w,b.h);});
  if(player){ctx.save();ctx.translate(player.x,player.y);ctx.rotate(player.angle);
ctx.fillStyle='#4dd'; ctx.fillRect(-player.w/2,-player.h/2,player.w,player.h); ctx.fillStyle='#0ff';
ctx.fillRect(8,-4,10,8); ctx.restore();}
  bullets.forEach(b=>{ctx.fillStyle='#ffeb7a'; ctx.fillRect(b.x-2,b.y-2,4,4);});
}

function loop(){update(); draw(); requestAnimationFrame(loop);}
loop();

window.addEventListener('resize',()=>{WIDTH=window.innerWidth;
HEIGHT=window.innerHeight; canvas.width=WIDTH; canvas.height=HEIGHT;});

</script>
</body>
</html>
```