BAB	: PEMROGRAMAN GAME
NAMA	: AFIF MUSYAYYIDIN
NIM	: 175150201111006
TANGGAL	: 05/11/2019
ASISTEN	: - DEDY RAKAYUDI PRASETYA
	- ZULFIKAR FAHMI FALAKH

TUGAS 1

A. Soal

1. Berdasarkan game yang telah dibuat, tambahkan code untuk mengatasi collision (tumbukan) antara tap/ click dari player terhadap zombie dan knight. Jika tap pada zombie, akan menambahkan score namun jika tap pada knight akan mengurangi life point.

2. Draw heart seperti yang ada pada gambar 8.2 sebagai indikator untuk life point. Jika life point berkurang satu, maka indikator heart akan berkurang satu. Indikator heart akan hilang semua jika life point menjadi nol. Muncul tulisan "GAME OVER" dan tombol restart yang jika ditekan akan mengulang game dari awal.

3. Draw tree di layer paling belakang, bergerak dari kanan ke kiri dengan velocity mengikuti level dari game.

4. Buat Tombol pause menghentikan game sementara.

B. Source Code

```
index.html

1   <html>
2
3   <head>
4       <title>Run Boy Run</title>
5       <link href="/favicon.ico" type="image/x-icon"
6   rel="icon" />
7       <link rel="stylesheet" type="text/css"
8   href="css/gamestyle.css" />
9   </head>
10
11  <body>
12      <div id="gameScreen">
13          <canvas id="cnv"></canvas>
14      </div>
15      <script src="js/gamescript.js"></script>
16  </body>
17
18  </html>
```

gamestyle.css

```css
1   #gameScreen {
2       position: relative;
3       background-image: url("../images/bg2.png");
4       background-size: 100%;
5       height: 360px;
6       width: 640px;
7       margin: 0 auto;
8   }
9
10  #cnv {
11      width: 100%;
12  }
13
14  #controller {
15      width: 100%;
16      padding: 10px;
17  }
```

gamescript.js

```javascript
1   (function() {
2       var knight,
3           knightImage,
4           level = 1,
5           velocity = 1,
6           numBush = 5,
7           numZombie = 3,
8           bush = [],
9           bushImage,
10          heart = [],
11          heartImage,
12          zombie = [],
13          score = 0,
14          life = 3,
15          gameOver = "",
16          btnRestart,
17          btnRestartImage,
18          animation = "",
19          isShowBtnRestart = false,
20          isBtnPauseClicked = false,
21          canvas,
22          btnPause,
23          btnPauseImage;
24
25      function gameLoop() {
26          animation =
27  requestAnimationFrame(gameLoop);
28          canvas.getContext("2d").clearRect(0, 0,
29  canvas.width, canvas.height);
30          var i;
31          btnPause.update();
32          btnPause.render();
33          knight.update();
34          knight.x -= level * velocity;
35          if (knight.x < -128) {
```

```
 36              knight.x = canvas.width +
 37    Math.round(Math.random() * canvas.width);
 38          }
 39          knight.render();
 40          for (i = 0; i < life; i++) {
 41              heart[i].update();
 42              heart[i].render();
 43          }
 44          for (i = 0; i < zombie.length; i++) {
 45              zombie[i].update();
 46              zombie[i].x -= level * velocity;
 47              zombie[i].render();
 48              if (zombie[i].x < -128) {
 49                  destroyZombie(zombie[i]);
 50                  setTimeout(spawnZombie, 1000);
 51                  life--;
 52              }
 53              if (score > level * 5) {
 54                  level++;
 55              }
 56          }
 57          for (i = 0; i < bush.length; i++) {
 58              bush[i].update();
 59              bush[i].x += velocity / 2;
 60              bush[i].render();
 61              if (bush[i].x > canvas.width + 65) {
 62                  bush[i].x = -80 -
 63    Math.floor(Math.random() * 3 + 1);
 64              }
 65          }
 66          if (life < 1) {
 67              life = 0;
 68              knight.w = 2640;
 69              gameOver = "Game Over";
 70              isShowBtnRestart = true;
 71              btnRestart.update();
 72              btnRestart.render();
 73              setTimeout(stopAnimation, 600);
 74          }
 75          drawHud();
 76      }
 77
 78      function sprite(options) {
 79          var that = {},
 80              frameIndex = 0,
 81              tickCount = 0,
 82              tickPerFrame = options.tickPerFrame ||
 83    0,
 84              numberOfFrame = options.numberOfFrame
 85    || 1;
 86          that.context = options.context;
 87          that.w = options.w;
 88          that.h = options.h;
 89          that.img = options.img;
 90          that.x = options.x;
 91          that.y = options.y;
 92          that.scaleRatio = 1;
 93          that.update = function() {
 94              tickCount += 1;
```

```
 95              if (tickCount > tickPerFrame) {
 96                  tickCount = 0;
 97                  if (frameIndex < numberOfFrame -
 98  1) {
 99                      frameIndex += 1;
100                  } else {
101                      frameIndex = 0;
102                  }
103              }
104          };
105          that.render = function() {
106              that.context.drawImage(
107                  that.img,
108                  frameIndex * that.w /
109  numberOfFrame,
110                  0,
111                  that.w / numberOfFrame,
112                  that.h,
113                  that.x,
114                  that.y,
115                  that.w / numberOfFrame,
116                  that.h
117              );
118          };
119          that.getFrameWidth = function() {
120              return that.w / numberOfFrame;
121          }
122          return that;
123      }
124
125      function stopAnimation() {
126          cancelAnimationFrame(animation);
127      }
128
129      function drawHud() {
130          var context = canvas.getContext("2d");
131          context.font = "bold 20px Consolas";
132          context.textAlign = "start";
133          context.fillStyle = "white";
134          context.fillText("Score: " + score,
135  canvas.width - 275, 50);
136          context.font = "bold 20px Consolas";
137          context.textAlign = "start";
138          context.fillStyle = "white";
139          context.fillText("Level: " + level,
140  canvas.width - 125, 50);
141          context.font = "bold 20px Consolas";
142          context.textAlign = "start";
143          context.fillStyle = "white";
144          context.fillText("Life: ", 30, 50);
145          context.font = "bold 70px Consolas";
146          context.textAlign = "center";
147          context.fillStyle = "#193439";
148          context.fillText(gameOver,
149  context.canvas.width / 2, context.canvas.height /
150  2 - 32);
151      }
152      canvas = document.getElementById("cnv");
153      canvas.width = 1024;
```

```
154        canvas.height = 460;
155        knightImage = new Image();
156        knight = sprite({
157            context: canvas.getContext("2d"),
158            w: 1740,
159            h: 210,
160            img: knightImage,
161            numberOfFrame: 10,
162            tickPerFrame: 5,
163            x: canvas.width,
164            y: canvas.height - 210
165        });
166        btnRestartImage = new Image();
167        btnRestart = sprite({
168            context: canvas.getContext("2d"),
169            w: 155,
170            h: 64,
171            img: btnRestartImage,
172            numberOfFrame: 1,
173            tickPerFrame: 1,
174            x: canvas.width / 2 - 70,
175            y: canvas.height / 2 - 10
176        });
177        btnPauseImage = new Image();
178        btnPause = sprite({
179            context: canvas.getContext("2d"),
180            w: 70,
181            h: 70,
182            img: btnPauseImage,
183            numberOfFrame: 0,
184            tickPerFrame: 0,
185            x: canvas.width / 2 - 30,
186            y: 15
187        });
188
189        function tap(e) {
190            var i,
191                loc = {},
192                dist,
193                distBtnRestart,
194                distBtnPause,
195                distKnight,
196                zombieToDestroy = [],
197                pos = getElementPosition(canvas),
198                tapX = e.targetTouches ?
199    e.targetTouches[0].pageX : e.pageX,
200                tapY = e.targetTouches ?
121    e.targetTouches[0].pageY : e.pageY,
122                canvasScaleRatio = canvas.width /
123    canvas.offsetWidth;
124            loc.x = (tapX - pos.x) * canvasScaleRatio;
125            loc.y = (tapY - pos.y) * canvasScaleRatio;
126            for (i = 0; i < zombie.length; i++) {
127                dist = distance({
128                    x: (zombie[i].x +
129    zombie[i].getFrameWidth() / 2 *
130    zombie[i].scaleRatio),
131
132
```

```
133           y: (zombie[i].y +
134    zombie[i].getFrameWidth() / 2 *
135    zombie[i].scaleRatio)
136            }, {
137                x: loc.x,
138                y: loc.y
139            });
140
141            if (dist < zombie[i].getFrameWidth() /
142    2 * zombie[i].scaleRatio) {
143                zombieToDestroy.push(zombie[i]);
144                console.log(zombie[i].x);
145            }
146        }
147        for (i = 0; i < zombieToDestroy.length;
148    i++) {
149            destroyZombie(zombieToDestroy[i]);
150            setTimeout(spawnZombie, 1000);
151        }
152        if (zombieToDestroy.length) {
153            score++;
154        }
155        distBtnRestart = distance({
156            x: (btnRestart.x +
157    btnRestart.getFrameWidth() / 2 *
158    btnRestart.scaleRatio),
159            y: (btnRestart.y +
160    btnRestart.getFrameWidth() / 2 *
161    btnRestart.scaleRatio)
162        }, {
163            x: loc.x,
164            y: loc.y
165        });
166        distBtnPause = distance({
167            x: (btnPause.x +
168    btnPause.getFrameWidth() / 2 *
169    btnPause.scaleRatio),
170            y: (btnPause.y +
171    btnPause.getFrameWidth() / 2 *
172    btnPause.scaleRatio)
173        }, {
174            x: loc.x,
175            y: loc.y
176        });
177        if ((distBtnRestart <
178    btnRestart.getFrameWidth() / 2 *
179    btnRestart.scaleRatio) && isShowBtnRestart) {
180            document.location.reload();
181        }
182        console.log(isBtnPauseClicked);
183        if ((distBtnPause <
184    btnPause.getFrameWidth() / 2 *
185    btnPause.scaleRatio) && !isBtnPauseClicked) {
186            cancelAnimationFrame(animation);
187            isBtnPauseClicked = true;
188            var context = canvas.getContext("2d");
189            context.font = "bold 50px Consolas";
190            context.textAlign = "center";
191            context.fillStyle = "#193439";
```

```
192              context.fillText("Game Paused",
193   context.canvas.width / 2, context.canvas.height /
194   2 - 32);
195          } else if ((distBtnPause <
196   btnPause.getFrameWidth() / 2 *
197   btnPause.scaleRatio) && isBtnPauseClicked) {
198              requestAnimationFrame(gameLoop);
199              isBtnPauseClicked = false;
200          }
201          distKnight = distance({
202              x: (knight.x + knight.getFrameWidth()
203   / 2 * knight.scaleRatio),
204              y: (knight.y + knight.getFrameWidth()
205   / 2 * knight.scaleRatio)
206          }, {
207              x: loc.x,
208              y: loc.y
209          });
210          if (distKnight < knight.getFrameWidth() /
211   2 * knight.scaleRatio) {
212              knight.x = canvas.width +
213   Math.round(Math.random() * canvas.width);
214              life--;
215          }
216      }
217
218      function destroyZombie(param) {
219          var i;
220          for (i = 0; i < zombie.length; i++) {
221              if (zombie[i] === param) {
222                  zombie[i] = null;
223                  zombie.splice(i, 1);
224                  break;
225              }
226          }
227      }
228
229      function spawnZombie() {
230          var zombieIndex,
231              zombieImage;
232          zombieImage = new Image();
233          zombieIndex = zombie.length;
234          zombie[zombieIndex] = sprite({
235              context: canvas.getContext("2d"),
236              w: 1740,
237              h: 210,
238              img: zombieImage,
239              numberOfFrame: 10,
240              tickPerFrame: Math.floor(Math.random()
241   * 8) + 5
242          });
243          if ((zombieIndex % 2) == 1) {
244              zombieImage.src =
245   "images/zombie_female_run.png";
246          } else {
247              zombieImage.src =
248   "images/zombie_male_run.png";
249          }
250
```

```
251        zombie[zombieIndex].x = canvas.width +
252  Math.random() * (canvas.width -
253  zombie[zombieIndex].getFrameWidth() *
254  zombie[zombieIndex].scaleRatio);
255        zombie[zombieIndex].y = canvas.height -
256  210;
257        zombie[zombieIndex].scaleRatio =
258  Math.random() * 0.5 + 0.5;
259      }
260
261    function spawnBush() {
262        var bushIndex,
262            bushImage;
263        bushImage = new Image();
264        bushIndex = bush.length;
265        bush[bushIndex] = sprite({
267            context: canvas.getContext("2d"),
268            img: bushImage,
269            w: 0,
270            h: 0,
271            x: 0,
272            y: 0,
273            numberOfFrame: 1,
274            tickPerFrame: 1
275        });
276        bush[bushIndex].x = 0 + Math.random() *
277  (canvas.width - bush[bushIndex].getFrameWidth() *
278  bush[bushIndex].scaleRatio);
279        if (bushIndex == 0) {
280            bush[bushIndex].w = 173;
281            bush[bushIndex].h = 66;
282            bush[bushIndex].y = canvas.height -
283  62;
284        }
285        if (bushIndex == 1) {
286            bush[bushIndex].w = 93;
287            bush[bushIndex].h = 47;
288            bush[bushIndex].y = canvas.height -
289  45;
290        }
291        if (bushIndex == 2) {
292            bush[bushIndex].w = 54;
293            bush[bushIndex].h = 55;
294            bush[bushIndex].y = canvas.height -
295  53;
296        }
297        if (bushIndex == 3) {
298            bush[bushIndex].w = 53;
299            bush[bushIndex].h = 76;
300            bush[bushIndex].y = canvas.height -
301  74;
302        }
303        if (bushIndex == 4) {
304            bush[bushIndex].w = 330;
305            bush[bushIndex].h = 301;
306            bush[bushIndex].y = canvas.height -
307  275;
308        }
309
```

```
310          bush[bushIndex].scaleRatio = Math.random()
311   * 0.5 + 0.5;
312          bushImage.src = "images/bush" + bushIndex
313   + ".png";
314       }
315
316     function spawnHeart() {
317         var heartIndex,
318             heartImage;
319         heartImage = new Image();
320         heartIndex = heart.length;
321         heart[heartIndex] = sprite({
322             context: canvas.getContext("2d"),
323             w: 50,
324             h: 40,
325             img: heartImage,
326             numberOfFrame: 1,
327             tickPerFrame: 1
328         });
329         if (heartIndex == 0) {
330             heart[heartIndex].x = 100;
331         }
332         if (heartIndex == 1) {
333             heart[heartIndex].x = 140;
334         }
335         if (heartIndex == 2) {
336             heart[heartIndex].x = 180;
337         }
338         heart[heartIndex].y = 22;
339         heart[heartIndex].scaleRatio =
340   Math.random() * 0.5 + 0.5;
341         heartImage.src = "images/life.png";
342     }
343
344     function getElementPosition(element) {
345         var parentOffset,
346             pos = {
347                 x: element.offsetLeft,
348                 y: element.offsetTop
349             }
350         if (element.offsetParent) {
351             parentOffset =
352   getElementPosition(element.offsetParent);
353             pos.x += parentOffset.x;
354             pos.y += parentOffset.y;
355         }
356         return pos;
357     }
358
359     function distance(p1, p2) {
360         var dx = p1.x - p2.x,
361             dy = p1.y - p2.y;
362
363         return Math.sqrt(dx * dx + dy * dy);
364     }
365     for (i = 0; i < numZombie; i++) {
366         spawnZombie();
367     }
368     for (i = 0; i < numBush; i++) {
```
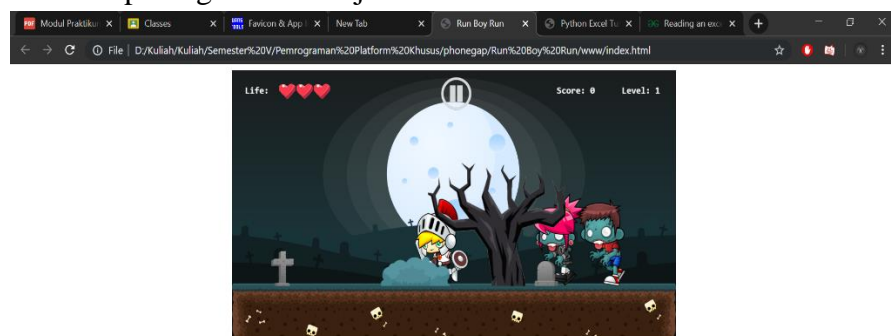
```
369          spawnBush();
370      }
371      for (i = 0; i < life; i++) {
372          spawnHeart();
373      }
374      gameLoop();
375      knightImage.src = "images/run-knight.png";
376      btnRestartImage.src = "images/restart.png";
377      btnPauseImage.src = "images/pause.png";
378      canvas.addEventListener("mousedown", tap);
379      canvas.addEventListener("touchstart", tap);
380  }());
```
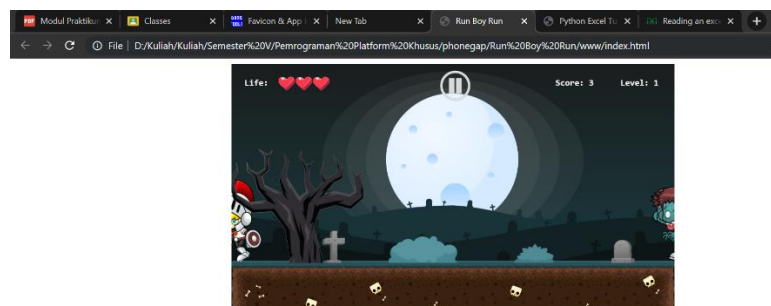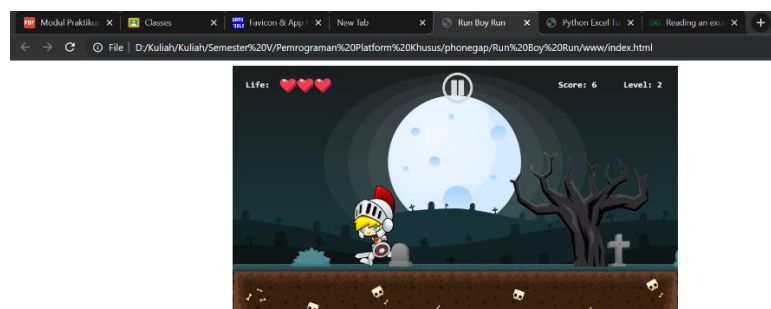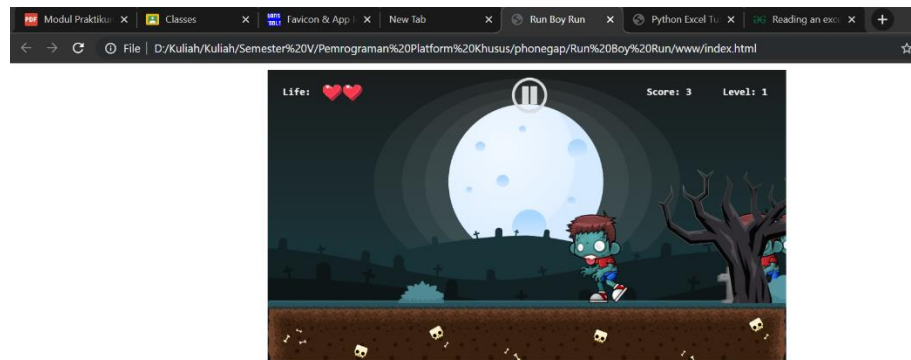
C. Screenshot

1. Tampilan game saat dijalankan

2. Tampilan game saat zombie diklik dan score bertambah
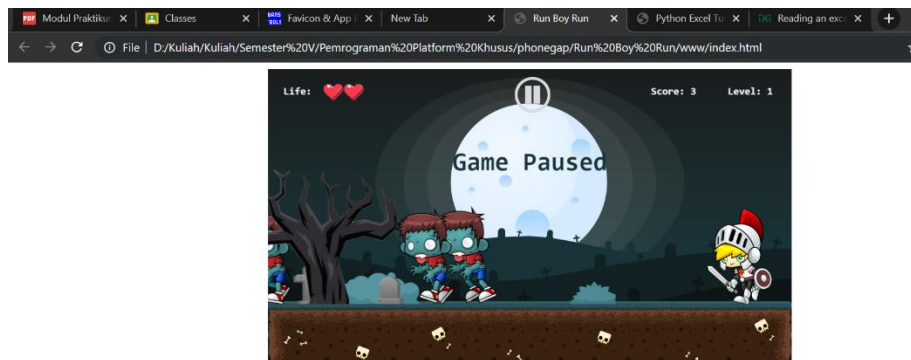

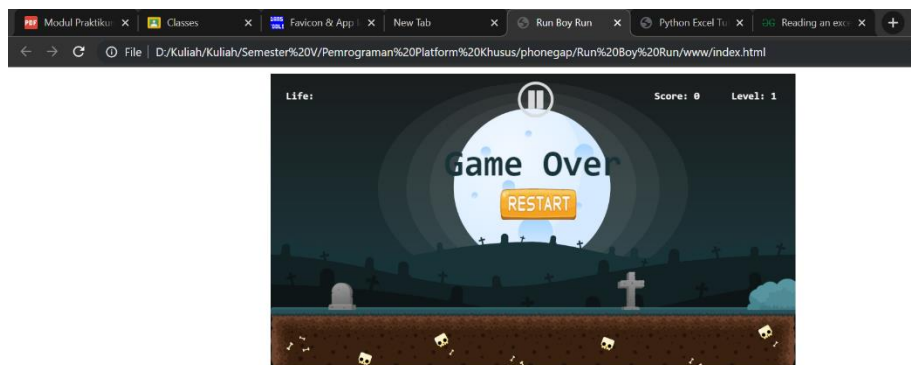3. Tampilan saat level pada game meningkat

4. Tampilan saat nyawa berkurang

5. Tampilan saat game di pause

6. Tampilan saat game berakhir

D. Penjelasan

1. Untuk mengatasi collision antara knight dan zombie saat player melakukan klik atau tap, maka dilakukan modifikasi pada script dengan menambahkan seleksi if pada function gameloop. Jika

knight yang diklik maka life akan berkurang dan sebaliknya jika zombie yang diklik maka score akan bertambah.

2. Untuk menambahkan life berupa gambar hati hal yang dilakukan adalah dengan menambahkan variabel baru berupa heart dan heratImage. Setelah itu ditambahkan method spawnHeart untuk melakukan update pada jumlah hati yang dimiliki pemain. Selanjutnya adalah melakukan perubahan pada drawHud untuk menampilkan gambar hati tersebut.

3. Disini dilakukan modifikasi pada jumlah variabel yang disimpan pada bushIndex untuk menambahkan asset berupa tree karena pada file images asset yang berupa tree/pohon tersebut bernama bush4.png. Setelah itu dilakukan penambahan seleksi if baru pada function spawnBush yang berisi ukuran dari asset pohon tersebut. Dengan begitu akan mencul asset baru berupa pohon yang bergerak dari kanan ke kiri sesuai dengan velocity pada game level.

4. Untuk menambahkan tombol pause ditambahkan variabel baru berupa btnPause, btnPuseImage, dan isBtnPauseClicked. Setelah itu dilakukan perubahan pada function tap dengan menambahkan seleksi if baru ketika button pause tersebut diklik maka animasi pada game akan terhenti dan tulisan "Game Paused" akan muncul.