

A collection of various colorful candies is scattered around the title. There is a blue round candy at the top, a yellow teardrop-shaped candy on the left, a brown candy with colorful sprinkles at the top right, an orange oval candy on the right, a red bean-shaped candy on the left, a green square candy at the bottom center, a purple star-shaped candy at the bottom right, and another brown candy with colorful sprinkles at the bottom left.

Candy World

Autor:

Andreea Sandu, 333AA

Cuprins

INTRODUCERE	3
PREZENTAREA TEMEI.....	3
OBIECTIVE	3
CANDY CRUSH.....	3
SUPORT TEHNIC	4
HTML	4
CSS	5
JAVASCRIPT	5
PROIECTE SIMILARE	6
ETAPA DE REALIZARE/IMPLEMENTARE	7
MOD DE UTILIZARE	13
CONCLUZII	15
BIBLIOGRAFIE	16

Introducere

Prezentarea temei

Candy World este un joc realizat într-o pagină web, folosind HTML, CSS și JavaScript. Ideea acestui joc a plecat de la celebrul “Candy Crush” și s-a încercat preluarea unor funcționalități. Jocul realizat are un număr infinit de niveluri și are ca scop strângerea cât mai multor puncte prin colectarea bomboanelor.

Obiective

În realizarea jocului au existat drept obiective:

- Existența unei pagini principale, pentru a porni jocul;
- Posibilitatea de a muta bomboanele prin tragere și plasare;
- Posibilitatea de a muta bomboanele doar în cazul în care se formează un grup de 3, 4 sau 5 bomboane;
- Primirea punctelor în cazul realizării formațiilor de 3, respectiv 4 bomboane;
- Apariția pentru o secundă a unei bomboane speciale în cazul realizării unei formații de 5, primirea a 64 de puncte (suma tuturor bomboanelor de pe tabla de joc), ștergerea bomboanelor existente și apariția unora noi;
- Existența unui număr diferit de mutări pentru fiecare nivel;
- Existența unui număr infinit de niveluri;
- Posibilitatea de a trece la nivelul următor;
- Posibilitatea de a restarta nivelul;

Candy Crush

Candy Crush este un joc creat de King pentru Facebook și a apărut în anul 2012. Ulterior au fost create versiuni pentru iOS, Android, Windows 10 și telefoanele cu Windows. Este considerat unul din primele jocuri gratuite de succes.

Jucătorii trebuie să interschimbe bomboane alăturate pentru a realiza potriviri de minim 3 bomboane de aceeași culoare. Prin crearea unui potriviri, bomboanele ce o formează sunt

eliminate, în locurile rămase goale vor cădea bomboanele de deasupra, iar în vârf apar altele generate în mod aleatoriu. În cazul creării unei potriviri cu mai mult de 3 elemente va apărea o bomboană specială.

Jocul are mai multe niveluri și la fiecare nivel există cerințe diferite, cum ar fi: realizarea unui scor dintr-un număr specificat de mutări, realizarea unui scor într-un interval de timp, colectarea unui număr de bomboane de o anumită culoare, eliminarea unor elemente de pe tabla de joc etc.

În cazul obținerii criteriilor necesare se poate trece la nivelul următor, în caz contrar se va pierde o viață. Epuizarea vieților presupune așteptarea unei perioade de timp pentru a primi vieți noi.

Cu toate că este un joc gratuit și se poate termina fără a fi nevoit să plătești, există opțiunea de a cumpăra diverse bomboane, acțiuni speciale sau vieți pentru a îți ușura jocul și a nu irosi timp. Datorită acestora compania a avut de câștigat aproape un milion de dolari zilnic în perioada de vârf.

Candy Crush Saga a obținut în anul 2016 premiul „People’s Choice Awards” pentru „Favourite Mobile Game 2016”. Un alt premiu câștigat este „International Mobile Gaming Awards” pentru „Best Social Game”.

Suport tehnic

HTML

HTML este limbajul standard pentru crearea paginilor Web.

Elementele utilizate sunt:

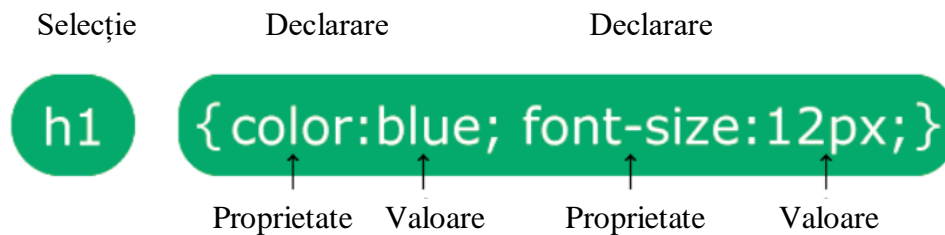
- `<!DOCTYPE html>` -> pentru a declara că documentul este de tipul HTML5;
- `<html>` -> elementul rădăcină al unei pagini HTML;
- `<head>` -> elementul conține metadatele despre pagina HTML;
- `<title>` -> elementul care specifică titlul paginii HTML;
- `<body>` -> elementul definește corpul documentului și conține tot conținutul vizibil;
- `` -> eticheta (tag) este un container în linie care marchează o parte a unui text/document;
- `<div>` -> eticheta care definește o secțiune într-o pagină HTML;

- `` -> eticheta utilizată pentru a introduce o imagine în pagina HTML;
- `<button>` -> eticheta care definește un buton apăsabil;
- `<script>` -> eticheta care este utilizată pentru a încorpora un script (în cazul de față JavaScript).
- `<link>` -> eticheta care definește relația dintre documentul curent și o resursă externă (în cazul de față un document css extern)

CSS

CSS este limbajul folosit pentru stilizarea unei pagini Web.

Exemplu de utilizare:



Selecție -> elementul HTML care va fi stilizat;

Declarare -> conține o proprietate și valoarea corespunzătoare;

JavaScript

JavaScript este limbajul de programare al paginilor Web.

Elemente utilizate:

- `var & const` -> variabile, cea de a doua definește o referință constantă la o valoare;
- `getElementByID()` -> metoda care returnează elementul care are atributul ID specificat;

- `querySelector()` -> metoda care returnează primul element care se potrivește cu selectorul CSS specificat din document;
- `.innerHTML` -> proprietate care setează sau returnează conținutul HTML al unui element;
- `function numeFunctie (parametri) { cod }` -> funcție JavaScript pentru executarea unui task;
- `createElement()` -> metoda care creează un element nod cu numele specificat;
- `setAttribute()` -> metoda care adaugă sau modifică atributul specificat elementului;
- `appendChild()` -> metoda care adaugă un nod ca copil al nodului specificat;
- `push()` -> metoda adaugă elemente la finalul matricei;
- `forEach()` -> metoda apelează o funcție pentru fiecare element al matricei;
- `every()` -> metoda care verifică dacă o funcție apelată pentru toate elementele matricei returnează adevărat;
- `includes()` -> metoda care determină dacă un string conține caracterele string-ului specificat;
- `setInterval()` -> metoda apelează o funcție la intervalul specificat;
- `clearInterval()` -> metoda șterge temporizarea setata prin `SetInterval()`.

Proiecte similare

Link-ul prezent în [1] conține proiectul de la care am pornit, un joc ce dorește să imite Candy Crush. Acest proiect prezintă modul în care se poate crea tabla de joc, se pot interschimba 2 bomboane și se poate verifica existența formațiilor de 3, respectiv 4 bomboane folosind JavaScript. Îmbunătățirile aduse de mine vor fi prezentate ulterior.

Etapa de realizare/implementare

1. Realizarea documentului HTML

Am început proiectul prin documentul HTML, care creează pagina Web.

Primul document HTML este pentru pagina de “START”. Am setat titlul și am realizat legătura cu documentul CSS, care se va ocupa de stilizare. Această pagină conține butonul “start”. În urma apăsării se va deschide pagina de joc.

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Candy Crush</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <form action="index.html" method="post" >
      <button id="b1" type="submit" name="start"><b style="font-family:Lucida Handwriting; font-style:italic
    </form>
  </body>
</html>
```

Fig. 1

Cel de-al doilea document HTML conține pagina de joc și are același “head”.

În corpul documentului HTML am adăugat:

- scorul, nivelul și numărul de mișcări folosind eticheta “span”;

```
<span class="score-board" style="color:black;font-family:Lucida Handwriting; font-style:italic; font-size:25px; font-weight:bold">
  Score
  <span id="score"></span>
</span>
<span class="level-board" style="color:white;font-family:Lucida Handwriting; font-style:italic; font-size:20px; font-weight:bold; position: absolute; right:10px">
  Level
  <span id="level"></span>
</span>
<br/><br/>
<span class="moves-board" style="color:white;font-family:Lucida Handwriting; font-style:italic; font-size:25px; font-weight:bold" >
  Moves
  <span id="moves"></span>
</span>
<br/><br/>
```

Fig. 2

- o secțiune în pagină, unde va apărea tabla de joc, folosind eticheta “div”;

```
<div class="grid" ></div>
```

Fig. 3

- cele 3 butoane: next (pentru trecerea la nivelul următor prin apelarea funcției nextLevel()), restart (pentru a restarta nivelul atunci când ne aflăm în cursul acestuia prin apelarea funcției restart()) și restart2 (pentru a restarta nivelul atunci când nu am obținut numărul necesar de puncte pentru a merge la nivelul următor);

```
<button id="next" type="button" onclick="nextLevel()" >Next level</button>
<button id="restart" type="button" onclick="restart()" >Restart</button>
<button id="restart2" type="button" onclick="restart2()" >Restart</button>
```

Fig. 4

- imagini pentru cele 6 stele care apar la finalul fiecărui nivel trecut;

```
</img>
</img>
</img>
</img>
</img>
</img>
```

Fig. 5

- legătura cu scriptul JavaScript.

```
<script src="app.js" > </script>
```

Fig. 6

2. Realizarea documentului CSS pentru stilizarea paginii Web

3. Realizarea scriptului JavaScript

Se vor prezenta funcțiile principale utilizate pentru realizarea jocului.

Funcția createBoard() creează tabla de joc prin adăugarea a 64 de elemente 'div' matricei squares. Acestor elemente li se atribuie o imagine în mod aleatoriu.

```
function createBoard(){
  for(let i=0; i<width*width;i++){
    const square= document.createElement('div')
    square.setAttribute('draggable',true)
    square.setAttribute('id',i)
    let randomColor= Math.floor(Math.random() * candyColors.length)
    square.style.backgroundImage=candyColors[randomColor]
    grid.appendChild(square)
    squares.push(square)
  }
}
```

Fig. 7

Funcția moveDown() găsește spațiile goale, coboară elementele de deasupra și generează noi bomboane în vârful.

```
function moveDown(){
  for(i=63;i>7;i--){
    if(squares[i].style.backgroundImage === ''){
      for(j=i;j>7;j-=width){
        squares[j].style.backgroundImage=squares[j-width].style.backgroundImage
      }
      let randomColor= Math.floor(Math.random() * candyColors.length)
      squares[j].style.backgroundImage=candyColors[randomColor]
    }
  }

  for(i=0;i<8;i++){
    if(squares[i].style.backgroundImage === ''){
      let randomColor= Math.floor(Math.random() * candyColors.length)
      squares[i].style.backgroundImage=candyColors[randomColor]
    }
  }
}
```

Fig. 8

Funcția `checkRowForFive()` caută 5 bomboane alăturate de aceeași culoare și în cazul în care găsește le elimină și adaugă o bomboană specială. În mod asemănător se realizează și funcția `checkColumnForFive()`.

```
function checkRowForFive(){
    for(i=0;i<60;i++){
        let rowOffive=[i,i+1,i+2,i+3,i+4]
        let decidedColor = squares[i].style.backgroundColor
        const isBlank= squares[i].style.backgroundColor === ''

        const notValid=[4,5,6,7,12,13,14,15,20,21,22,23,28,29,30,31,36,37,38,39,44,45,46,47,52,53,54,55]
        if(notValid.includes(i)) continue

        if(rowOffive.every(index => squares[index].style.backgroundColor === decidedColor && !isBlank)){
            score+=5
            scoreDisplay.innerHTML=score
            rowOffive.forEach(index => {
                squares[index].style.backgroundColor=''
            })
            squares[i].style.backgroundColor='url(poze/bomb.png)'
        }
    }
}
```

Fig. 9

Funcțiile `checkRowForFour()`, `checkColumnForFour()`, `checkRowForThree()`, `checkColumnForThree()` se realizează în aceeași manieră, lipsind doar adăugarea bomboanei speciale.

Funcția `checkForBomb()` caută bomboana specială și când o găsește adaugă 64 de puncte, golește toată tabla și apoi generează noi bomboane.

```
function checkForBomb(){
    for(i=0;i<64;i++){
        if(squares[i].style.backgroundColor === 'url("poze/bomb.png")' ){
            setTimeout(function(){
                emptyBoard()
                fillingBoard(),1000)
            score+=64
            scoreDisplay.innerHTML=score
        }
    }
}
```

Fig. 10

Funcția emptyBoard() golește toată tabla de joc.

```
function emptyBoard() {  
    for(i=0;i<64;i++){  
        squares[i].style.backgroundImage = ''  
    }  
}
```

Fig. 11

Funcția fillingBoard() generează în mod aleatoriu o bomboană pentru fiecare loc din tabla de joc.

```
function fillingBoard(){  
    for(i=0;i<64;i++){  
        let randomColor= Math.floor(Math.random() * candyColors.length)  
        squares[i].style.backgroundImage=candyColors[randomColor]  
    }  
}
```

Fig.12

Funcția stop() verifică dacă s-au epuizat mutările și când rămânem fără mutări se va șterge intervalul la care se repetau funcțiile. În funcție de scor va apărea butonul de „next level” sau de “restart”. În cazul unui scor care ne permite să trecem la următorul nivel vor apărea pe ecran scorul și stelele obținute.

```
function stop(){  
    if(moves===0){  
        if(score>(100+50*level)){  
            clearInterval(myVar)  
            emptyBoard()  
            document.getElementById("score-level").style.visibility="visible"  
            scoreNext.innerHTML=score  
            if(score>=100+50*level+200){  
                document.getElementById("star1y").style.visibility="visible"  
                document.getElementById("star2y").style.visibility="visible"  
                document.getElementById("star3y").style.visibility="visible"  
                document.getElementById("next").style.visibility= "visible"  
            }  
            else if(score>=100+50*level+100){  
                document.getElementById("star1y").style.visibility="visible"  
                document.getElementById("star2y").style.visibility="visible"  
                document.getElementById("star3g").style.visibility="visible"  
                document.getElementById("next").style.visibility= "visible"  
            }  
            else{  
                document.getElementById("star1y").style.visibility="visible"  
                document.getElementById("star2g").style.visibility="visible"  
                document.getElementById("star3g").style.visibility="visible"  
                document.getElementById("next").style.visibility= "visible"  
            }  
        }  
        else{  
            clearInterval(myVar)  
            emptyBoard()  
            document.getElementById("restart").style.visibility= "visible"  
        }  
    }  
}
```

Fig. 13

Pentru a se repeta apelarea funcțiilor am folosit `window.setInterval()`.

```
var myVar =window.setInterval(function() {
    moveDown()
    stop()
},100)

window.setInterval(function() {

    checkRowForFive()
    checkColumnForFive()
    checkRowForFour()
    checkColumnForFour()
    checkRowForThree()
    checkColumnForThree()

},500)
```

Fig. 14

Funcția `nextLevel()` ascunde scorul și stelele obținute la nivelul anterior, incrementează nivelul, resetează scorul, stabilește numărul de mutări și creează un nou interval.

```
function nextLevel() {
    document.getElementById("star1y").style.visibility= "hidden"
    document.getElementById("star1g").style.visibility= "hidden"
    document.getElementById("star2y").style.visibility= "hidden"
    document.getElementById("star2g").style.visibility= "hidden"
    document.getElementById("star3y").style.visibility= "hidden"
    document.getElementById("star3g").style.visibility= "hidden"
    document.getElementById("next").style.visibility= "hidden"
    document.getElementById("score-level").style.visibility="hidden"
    level=level+1
    score=0
    moves=(15+level*5)
    scoreDisplay.innerHTML=score
    movesDisplay.innerHTML=moves
    levelDisplay.innerHTML=level
    var myVar =window.setInterval(function() {
        moveDown()
        stop()
    },100)
}
```

Fig. 15

Funcția `restart()` se realizează în mod asemănător, mai puțin incrementarea nivelului.

Îmbunătățirile aduse de mine proiectului de la care am plecat constau în:

- realizarea unei funcții care să coboare bomboanele când se găsesc locuri libere și să genereze noi bomboane în vârf;
- realizarea unei funcții care să golească tabla de joc;
- realizarea unui funcții care să umple tabla de joc cu bomboane alese în mod aleatoriu;
- realizarea unei funcții care să verifice prezența formațiilor de 5 bomboane și să genereze o bomboană specială;
- realizarea unei funcții care să verifice dacă există pe tabla de joc o bomboană specială și în acest caz să golească tabla de joc și să genereze noi bomboane;
- realizarea unei funcții care atunci când rămâi fără mutări, în funcție de scor poți trece la nivelul următor sau relua nivelul;
- realizarea unei funcții pentru a trece la nivelul următor;
- realizarea uni funcții pentru a reseta nivelul.

Mod de utilizare

Pagina Web se deschide cu pagina de start. Pentru a începe se apasă butonul “START”.

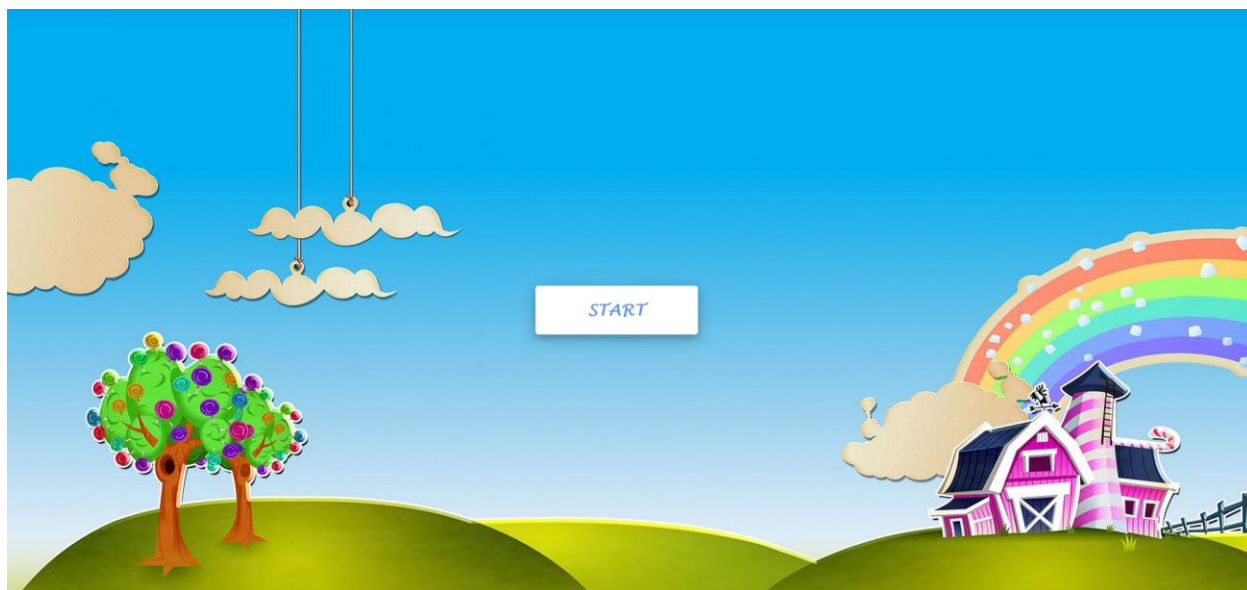


Fig. 16

După apăsarea butonului “START” se va deschide pagina de joc care afișează scorul, numărul de mutări, nivelul, tabla de joc și butonul “Restart”. Prin apăsarea butonului “Restart” se va reseta nivelul.

Jocul constă în interschimbarea bomboanelor alăturate pentru a crea formații de aceeași culori și a primi puncte.

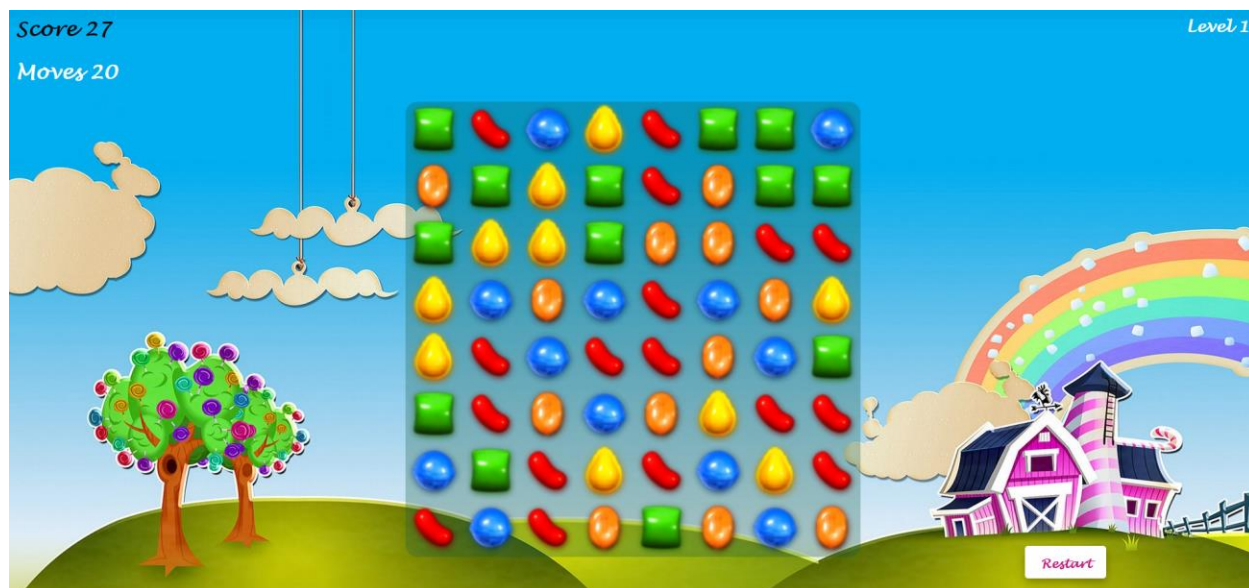


Fig. 17

Când se epuizează numărul de mutări va apărea următoarea pagină, unde poți vedea numărul de stele pe care l-ai obținut, scorul și butonul pentru a trece la nivelul următor.

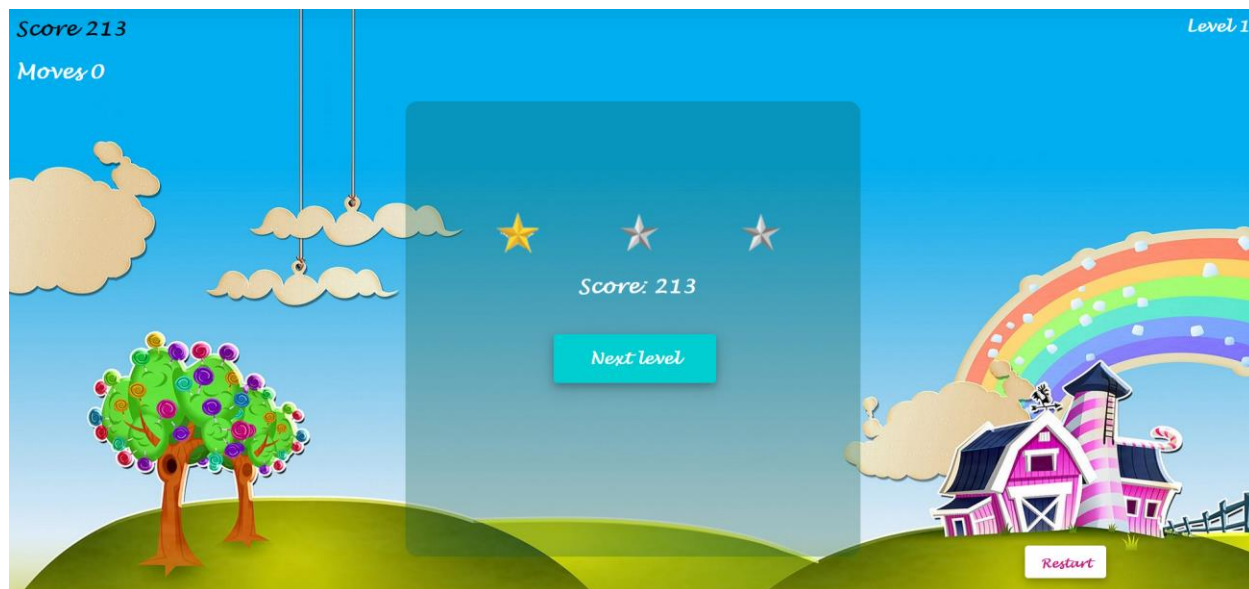


Fig. 18

În cazul în care nu s-a obținut scorul necesar va apărea următoarea pagină, pentru a relua nivelul.

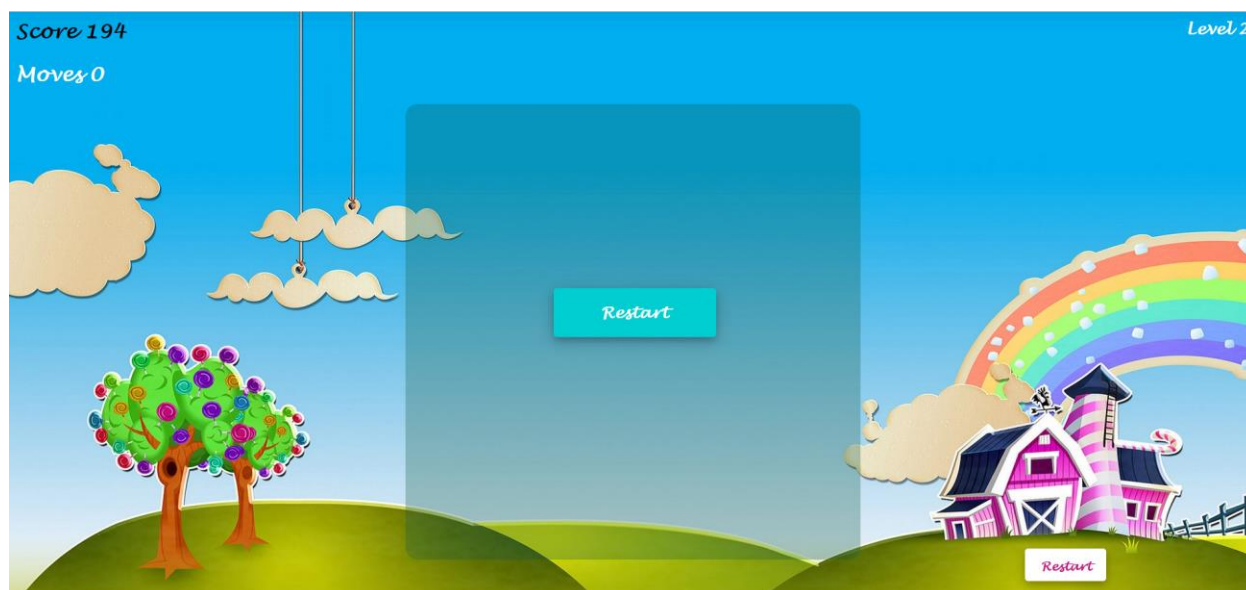


Fig. 19

Jocul nu are un final, existând un număr nelimitat de niveluri.

Concluzii

Obiective

Am reușit să îndeplinesc toate obiectivele propuse. Cu toate acestea s-ar putea face mult mai multe îmbunătățiri, cum ar fi:

- apariția unei bomboane speciale și la realizarea unei formații de 4 bomboane de aceeași culoare, care ar putea goli doar un rând sau o coloană;
- efecte vizuale la utilizarea bomboanelor speciale;
- criterii suplimentare pentru a merge la nivelul următor etc.

Utilitate

Jocul poate fi utilizat atât pentru divertisment, cât și pentru a îți testa răbdarea și atenția la detalii și perspicacitatea (alegerea celor mai bune mutări).

Bibliografie

1. Build your own CANDY CRUSH using JavaScript, HTML and CSS | Ania Kubow (2020)
<https://www.youtube.com/watch?v=XD5sZWxwJUk>
2. Candy Crush Saga
https://en.wikipedia.org/wiki/Candy_Crush_Saga
3. W3schools
<https://www.w3schools.com/>