

# JEGYZŐKÖNYV

Web technológia alapjai

GameDev weboldal

Készítette: **Danyi Károly**

Neptunkód: **RDC1L4**

Dátum: 2024.május.05

Bevezetés .....	3
Fájl Lista .....	4
HTML fájlok: .....	4
CSS fájlok: .....	4
JS fájlok .....	4
JSON fájlok .....	4
Egyéb fájlok .....	4
Index.html .....	5
games.html .....	7
Táblázat formázás: .....	9
Form.html .....	10
Form JS .....	11
Engine.html .....	14
JQuery és js Animáció .....	15
Kód magyarázat animációhoz.....	16
doc.html.....	17
Szorgalmi .....	17

## Bevezetés

Üdvözljük a GameDev weboldal dokumentációjában! Ez a projekt egy olyan online platformot céloz meg, amely a játékfejlesztés világába nyújt betekintést és segítséget nyújt mindazoknak, akik érdeklődnek a játékfejlesztés iránt. A weboldal célja, hogy összegyűjtse és bemutassa a legfontosabb információkat a különböző gametípusokról és game engine-ekről, valamint segítséget nyújtson azoknak, akik szeretnék elkezdni a saját játékok fejlesztését. A gametípusok és game engine-ek áttekintése fontos része ennek a dokumentációnak. A gametípusok olyan kategóriák, amelyekbe a játékok csoportosíthatók, és különböző jellemzők alapján különíthetők el egymástól. Ezek a kategóriák segítenek megérteni a különféle játékstílusokat és műfajokat, amelyekre a fejlesztők ötleteiket építhetik. A game engine-ek pedig olyan szoftveres eszközök, amelyek lehetővé teszik a fejlesztők számára, hogy játékokat hozzanak létre, megkönnyítve ezzel a grafikai, fizikai és egyéb technikai kihívások kezelését. A dokumentáció további részeiben részletesen bemutatjuk a különböző gametípusokat és game engine-eket, valamint útmutatást nyújtunk azoknak, akik saját játékfejlesztési projektbe szeretnének kezdeni. Reméljük, hogy ez a weboldal inspirációt és segítséget nyújt mindazoknak, akik érdeklődnek a játékfejlesztés iránt, és lehetőséget biztosít arra, hogy saját ötleteiket valóra váltsák a digitális játékok világában

# Fájl Lista

## HTML fájlok:

- index.html
- form.html
- engine.html
- games.html
- doc.html

## CSS fájlok:

- gamesstyles.css
- styles.css
- formstyleless.css
- stylesJQ.css

## JS fájlok

- RDC1L4\_1.js
- server.js

## JSON fájlok

- package.json
- Json.json

## Egyéb fájlok

- RDC1L4.pdf
- Gamemakerlogo.png
- godot-logo.png
- Unity-logo.jpg
- Unreal\_Engine.png
- StardewV.jpg
- Terraria.jpg
- StardewValleyTrailer.mp4
- TerrariaTrailer.mp4

## Index.html

### Weboldal Szerkezete:

A GameDev weboldal HTML, CSS és JavaScript nyelveken épül fel.

### Fejléc (Header):

- A fejlécben található a navigációs sáv, amely lehetővé teszi a felhasználók számára, hogy könnyen elnavigáljanak az oldal különböző részei között.
- A navigációs sáv tartalmazza a következő linkeket: Főoldal, Játékok, Fejlesztő környezetek, Tipp kérő és Dokumentum.

### Fő Tartalom (Main):

- A fő tartalom rész három fő szakaszból áll: "Bemutató", "Pixel Art Játék" és "2D Játék". A "Bemutató" szakasz bemutatja a GameDev weboldalt és annak főbb jellemzőit.
- A "Pixel Art Játék" és "2D Játék" szakaszok részletesen ismertetik ezeket a játékfajtákat, jellemzőiket és példákat mutatnak be.

### Videó Bemutató (Video):

- A videó bemutató szakaszban található egy beágyazott videólejátszó, amely lehetővé teszi a felhasználók számára, hogy megnézzék a kiválasztott játékok trailerjeit.
- A videólejátszó alatt gombok találhatók, amelyek segítségével a felhasználók váltogathatnak a rendelkezésre álló videók között, lejátszhatják, szüneteltethetik vagy újraindíthatják azokat.

### Lábjegyzet (Footer):

- A lábjegyzet tartalmazza a weboldal szerzőjének nevét és egyéb dolgokat.

### Technikai Megvalósítás:

- A weboldal HTML5, CSS3 és JavaScript segítségével készült.
- A Bootstrap keretrendszert használja a responszív elrendezés és a stílusok kezeléséhez.
- A videólejátszóhoz szükséges funkciókat JavaScript segítségével valósították meg.

```
script>
var videoElement = document.getElementById("gameVideo");
var videos = ["StardewValleyTrailer.mp4", "TerrariaTrailer.mp4"];
function playPause() {
    if (videoElement.paused) {
        videoElement.play();
    } else {
        videoElement.pause();
    }
}

function restartVideo() {
    videoElement.currentTime = 0;
    videoElement.play();
}

function playVideo(source) {
    videoElement.src = source;
    videoElement.play();
}

var currentVideoIndex = 0;

videoElement.addEventListener('ended', function() {
    currentVideoIndex = (currentVideoIndex + 1) % videos.length;
    videoElement.src = videos[currentVideoIndex];
    videoElement.play();
});
/script>
```

## games.html

### Weboldal Szerkezete:

A Games weboldal HTML és CSS nyelveken épül fel.

### Fejléc (Header):

- A fejlécben található a GameDev logó és a navigációs sáv, amely lehetővé teszi a felhasználók számára, hogy könnyen elnavigáljanak az oldal különböző részei között.
- A navigációs sáv tartalmazza a következő linkeket: Főoldal, Játékok, Fejlesztő környezetek, Tipp kérő és Dokumentum.

### Fő Tartalom (Main):

- A fő tartalom rész kettő fő szakaszból áll: "Játék típusok bemutató" és "Táblázat" részből.

### Bemutató:

- Különböző játéktípusokat mutat be, nagyon látványos CSS formázással és interaktív elemekkel.

### Táblázat:

- Egy játékfejlesztési táblázat, amely különböző játéktípusokat sorol fel, és ezeknek a típusoknak a fejleszthetőségét, népszerűségét, grafikai komplexitását és játékmenet bonyolultságát mutatja be. Többnyire irányadónak van tervezve.

### CSS:

- A CSS tartalmazza az alap beállításait az oldalnak.
- Tartalmazza a kártyák teljeskörű animációit, az adott oldalon.
- A html kódban és a css fájlban is van CSS formázás.
- Tábla stílusának formázását is tartalmazza

Kódrészlet:

Kártya animációk részlet:

```
.wrap {
display: flex;
flex-wrap: nowrap;
justify-content: space-between;
width: 85vmin;
height: 65vmin;
margin: 2rem auto;
border: 8px solid;
border-image: linear-gradient(
-50deg,
green,
#00b300,
forestgreen,
green,
lightgreen,
#00e600,
green
)
1;
transition: 0.3s ease-in-out;
position: relative;
overflow: hidden;
}
```

```
.overlay {
position: relative;
display: flex;
width: 100%;
height: 100%;
padding: 1rem 0.75rem;
background: #186218;
transition: 0.4s ease-in-out;
z-index: 1;
}
.overlay-content {
display: flex;
flex-direction: column;
justify-content: space-between;
width: 15vmin;
height: 100%;
padding: 0.5rem 0 0 0.5rem;
border: 3px solid;
border-image: linear-gradient(
to bottom,
#aea724 5%,
forestgreen 35% 65%,
#aea724 95%
)
0 0 0 100%;
transition: 0.3s ease-in-out 0.2s;
z-index: 1;
}
```

A fenti kód, csak egy részlet, de már ebből is látható hogy nagyon komplex az animációknak a beállítása és a kártyák stílusának szerkesztése. Ez mind a jó megjelenés érdekében került bele.



## Táblázat formázás:

```
table {  
  width: 100%;  
  border-collapse: collapse;  
  border: 2px solid #336633;  
  margin-bottom: 20px;  
}  
  
th, td {  
  padding: 12px;  
  text-align: left;  
  border-bottom: 1px solid #336633;  
}  
  
th {  
  background-color: #336633;  
  color: white;  
  font-weight: bold;  
  text-transform: uppercase;  
}
```

Tartalmazza az alapvető beállításokat. Egyszerű formázás, a letisztultság érdekében.

TÍPUS	FEJLESZTHETŐSÉG (S-F)	NÉPSZERŰSÉG (1-5)	GRAFIKA KOMPLEXITÁSA (1-5)	JÁTÉKMENET BONYOLULTSÁGA (1-5)
Akcio	B	5	4	3
Souls-like	A	4	5	4
Kaland	B	5	5	3
Szerepjáték	C	5	4	4
Stratégiai	C	3	3	5
Competitive	A	4	3	2
Szimuláció	C	3	4	3
Horror	D	4	4	3
Platformer	B	4	3	4
Sport	B	3	3	3
Indie	C	3	2	3
MMO	F	5	5	5

Ez a CSS kód felel a táblázat megjelenéséért. Beállítja a táblázat szélességét 100%-ra, hogy teljesen kitöltse a rendelkezésre álló teret. A táblázat körvonalát egy vékony, zöld színű vonal határozza meg. A cellák belső térközét, az igazítást és a vonalakat állítja be, hogy tisztábbá tegye a táblázatot. A táblázat fejlécének háttérszíne zöld lesz, hogy kiemelje az elnevezéseket, és a szöveg fehér színű lesz, hogy jól látható legyen a zöld háttéren. Egyszerű, de hatékony stílusokat alkalmaz a táblázat megjelenítésére.

## Form.html

Ez egy HTML kód, amely egy űrlapot hoz létre, amelyet "Tipp kérő"-nek neveznek. Az űrlapon a felhasználók játékokkal kapcsolatos információkat tudnak megosztani.

Az űrlapnak több mezője van, mint például:

- játékcím,
- játék leírása,
- játéktípus,
- platformok,
- fejlesztő környezet,
- játék színe ,
- kiadás dátuma.

Ezeket a mezőket megfelelően formázza és validálja a JavaScript kód. A weboldalnak van egy navigációs sávja is, amely lehetővé teszi a felhasználók számára, hogy könnyen navigáljanak más oldalakra a webhelyen.

Ezenkívül van egy AJAX hívás is, amely egy JSON fájlt tölt be és megjeleníti a weboldalon található játékok adatait.

Összességében ez az HTML kód egy felhasználóbarát űrlapot és navigációs sávot kínál a felhasználóknak, hogy megoszthassák és böngészhessék a játékokkal kapcsolatos információkat.

## Form JS

A JavaScript kód egy `validateForm` nevű függvényt definiál, amelyet az űrlap validálására használnak. A függvény ellenőrzi az űrlap különböző mezőinek tartalmát, például a játékcímet, a leírást, a típust, a kiválasztott platformokat és fejlesztő környezeteket, valamint a kiadás dátumát. Ha bármelyik mező nem felel meg a validációs követelményeknek (például üres vagy nem megfelelő formátumú), akkor a függvény beállítja a megfelelő hibaüzeneteket és visszatérési értéként `false`-t ad vissza, jelezve, hogy az űrlap nem lett sikeresen validálva.

Ha minden mező megfelel a követelményeknek, akkor a függvény visszatérési értéke `true`, és egy sikeres küldés üzenetet jelenít meg.

A kód továbbá tartalmaz még néhány segédmetódust, például a `changeColor`, amely az űrlap egyes elemeinek háttérszínét változtatja a felhasználó által kiválasztott érték alapján, valamint a `resetColors`, amely visszaállítja az űrlap elemeinek stílusát és törli a hibaüzeneteket.

Végül a kód tartalmaz egy AJAX hívást is, amely egy JSON fájlt tölt be a játékok adataival, és ezeket megjeleníti a weboldalon. Ez lehetővé teszi a felhasználók számára, hogy böngésszenek és megtekintsék az eddigi bevitt adatokat.

```
$.ajax({
  url: 'json.json', // A JSON fájl elérési útja
  dataType: 'json',
  success: function (data) {
    // Az adatok megjelenítése
    var jatekokDiv = $('#jatekok');
    $.each(data.jatekok, function (index, jatek) {
      var jatekHtml = `
        <div>
          <h2>${jatek.cim}</h2>
          <p><strong>Leírás:</strong> ${jatek.leiras}</p>
          <p><strong>Típus:</strong> ${jatek.tipus}</p>
          <p><strong>Platformok:</strong> ${jatek.platformok.join(', ')}</p>
          <p><strong>Környezet:</strong> ${jatek.kornyezet}</p>
          <p><strong>Szín:</strong> <span style="background-color: ${jatek.szin};">${jatek.szin}</span></p>
          <p><strong>Kiadás dátuma:</strong> ${jatek.kiadas_datuma}</p>
        </div>
      `;
      jatekokDiv.append(jatekHtml);
    });
  },
  error: function (xhr, status, error) {
    console.error('Hiba történt a JSON fájl betöltése közben:', error);
  }
});
```

A képen egy szövegszerkesztőben megjelenített kódrészlet látható, amelynek szintaxiskiemelése van. A kód JavaScriptben íródott, és a jQuery nevű JavaScript könyvtárat használja egy AJAX kérés végrehajtásához. Az AJAX kérés JSON adatokat kér le egy meghatározott URL-ről. A kódban magyar nyelvű kommentek találhatók, amelyek leírják a különböző részeket, például az AJAX hívás inicializálását és a siker és hiba kezelését. A siker esetén HTML elemek dinamikusan létrejönnek és adatokkal lesznek feltöltve a JSON válaszból.

```

function validateForm() {

    var regex = /^[A-Za-z0-9\s]+$/;

    var title = document.getElementById("gameTitle").value;
    var description = document.getElementById("gameDescription").value;
    var type = document.getElementById("gameType").value;
    var platforms = document.querySelectorAll('input[name="platform"]:checked');
    var asd = document.querySelectorAll('input[name="radiobutton"]:checked');
    var color = document.getElementById("gameColor").value;
    var releaseDate = document.getElementById("releaseDate").value;

    var isValid = true;

    if (title == "" || !regex.test(title)) {
        document.getElementById("gameTitle").style.border = "2px solid red";
        document.getElementById("gameTitleError").style.color = "red";
        var labelElement = document.getElementById("gameTitleError");
        labelElement.innerHTML = "Hibás Adat";

        isValid = false;
    } else {
        document.getElementById("gameTitleError").style.color = "green";
        document.getElementById("gameTitle").style.border = "2px solid green";
        var labelElement = document.getElementById("gameTitleError");
        labelElement.innerHTML = "<strong>Játékcím:</strong>";
    }

    if (description == "") {
        document.getElementById("gameDescription").style.border = "2px solid red";
        document.getElementById("gameDescriptionError").style.color = "red";
        var labelElement = document.getElementById("gameDescriptionError");
        labelElement.innerHTML = "Hibás Adat";
        isValid = false;
    }
}

```

A függvény különböző bemeneti mezőket, például a 'gameTitle', 'releaseDate' és 'platforms' mezőket ellenőrzi az értékeikre, és feltételes formázást alkalmaz annak alapján, hogy a bemenetek megfelelnek-e bizonyos kritériumoknak. Például, ha a 'gameTitle' mező üres, akkor a hozzá tartozó címkét pirosra változtatja és hibaüzenetet jelenít meg.

## Engine.html

A weboldal fő részében négy fejlesztői környezetet mutat be:

- Unity,
- Unreal Engine,
- Godot Engine,
- GameMaker Studio.

Mindegyik fejlesztői környezethez tartozik egy gomb, amelyek segítségével további információkhoz juthatunk.

Minden fejlesztői környezetet egy kép és egy rövid leírás kíséri, ami segít megérteni, hogy miért népszerű és milyen lehetőségeket kínál a játékfejlesztőknek.

A leírások az adott fejlesztői környezet erősségeire és felhasználóbarát jellegére fókuszálnak, és könnyen érthető módon fogalmazzák meg az információkat.

A weboldal stílusa egyszerű és letisztult, a gombok és a szövegek könnyen észrevehetők és érthetők.

A képek és a színek segítségével vonzóvá teszik az oldalt és segítenek az információk megjelenítésében. A navigációs sáv és a gombok áttekinthetővé teszik az oldalt, így a felhasználók könnyen eligazodhatnak és megtalálhatják az érdeklődésüknek megfelelő információkat.

## JQuery és js Animáció

Ezen a weboldalon található szinte összes animáció, JQuery és js-el van megoldva.

Amikor az oldalon a "Unity", "Unreal Engine", "Godot Engine" vagy "GameMaker Studio" gombra kattintunk, akkor azokhoz tartozó részek megjelenése megváltozik animációval.

Például a gombra kattintva az adott rész elmozdul és nő, majd újra kattintva visszatér az eredeti méretéhez és helyzetéhez.

Emellett egy másik gombra kattintva az oldalsáv megnyitható vagy bezárható, ami további interaktivitást tesz lehetővé az oldalon.

Ezen kívül négy másik gomb lehetővé teszi az információk megjelenítését vagy elrejtését a különböző fejlesztői környezetekhez tartozó leírásokban.

Ez a kód segít a felhasználónak az oldalon való navigálásban és az információk könnyű megjelenítésében és elrejtésében az adott témákhoz kapcsolódóan.

## Kód magyarázat animációhoz

```
$("#game2").click(function () {  
  if (!isrOpen4) {  
    $("#gamediv2").animate({ left: '700px', width: '400px', height:'500px', fontSize: '12pt'}, 1000);  
    isrOpen4 = true;  
  } else {  
    $("#gamediv2").animate({ left: '100%', width: '-250px', height:'0px', fontSize: '12pt' }, 1000);  
    isrOpen4 = false;  
  }  
});
```

Gombra kattintáskor megnézi a jelenlegi helyét a div-nek, egy boolean típusú változóval. Az adott helytől függ hogy melyik animáció megy végbe.

```
$("#oldalsav").click(function () {  
  if (!OldalisOpen) {  
    $('.sidePanel').toggleClass('show');  
    $('.sidePanel2').toggleClass('show');  
    OldalisOpen = true;  
  } else {  
    $('.sidePanel').removeClass('show');  
    $('.sidePanel2').removeClass('show');  
    OldalisOpen = false;  
  }  
});
```

Az oldalsávokat lehet kinyitni vele és bezárni. Ezt mind egy gomb kattintással. Itt is az előző technikát is alkalmazzuk.

```
$("#ub").click(function(){  
  $("#ut").toggle(1000);  
});  
  
$("#unb").click(function(){  
  $("#unt").toggle(1000);  
});  
  
$("#godb").click(function(){  
  $("#godt").toggle(1000);  
});  
  
$("#gmb").click(function(){  
  $("#gmt").toggle(1000);  
});
```

Itt <p> tagokat változtatunk. Ezek tartalmazzák a rövid leírását a játékfejlesztő engine/környezeteknek.



## doc.html

Ez a weboldal a dokumentációját tartalmazza a projektnek, egy beágyazott dokumentummal. A dokumentum leírja hogy miket tartalmaznak az oldalak, milyen fájlok szerepelnek a projektben és hogy hogyan épülnek fel ezek a fájlok.

```
<div style="width: 100%; height: 1000px;">
  <embed src="RDC1L4.pdf" type="application/pdf" width="100%" height="100%"/>
</div>
```

Itt <embed> taggal beágyazzuk a fájlt a weboldalba. Típuskén meglehet adni milyen fájlról van szó.

## Szorgalmi

Server.js:

```
const express = require('express');
const path = require('path');

const app = express();
const port = 3000;

app.use(express.static(path.join(__dirname)));

app.get('/', (req, res) => {
  res.sendFile(path.join(__dirname, 'main', 'index.html'));
});

app.listen(port, () => {
  console.log(`A szerver fut a http://localhost:${port} címen.`);
});
```

A localhost: “3000”-es porton való megjelenítésre node js konfigurálására került sor. Köztük “npm” kódok kerültek beírásra a terminálba. Elindításhoz “npm start” parancs beírása szükséges.

Package.json tartalma:

```
{
  "name": "rdc114webtech",
  "version": "1.0.0",
  "description": "beadando",
  "main": "server.js",
  "scripts": {
    "start": "node server.js"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.19.2"
  }
}
```

Ezt egy npm paranccsal automatikusan létre lehet hozni.

A működéshez még le kellett tölteni a node.js-t és node modulokat.

Itt látható egy minta, hogy megjelenik az index.html, css-el is a localhost:3000-en.

