**Projekt neve:**

Activity Webition

Digitalizált asztali társasjáték  
  
**Fejlesztette**:

Ádám Róbert

**Tartalom**

[**Bevezetés**](#_l84e9va0ufq1) **3**

[Célcsoport](#_a5xbp0z3t7ty) 3

[Projektcélok](#_8wj5ui2wczk) 3

[**Felhasznált technológiák**](#_z0j8nru0nh89) **4**

[**Rendszerkövetelmények**](#_4wbpx6b76bjp) **4**

[Minimális szoftverkövetelmények](#_l5lu5strmea7) 4

[**Architektúra**](#_gjbmbywxi77d) **5**

[**Használt szoftverek és funkcióik**](#_ajmj0rorgn0q) **5**

[**Adatbázis**](#_9i0lkh7wpg2w) **6**

[Táblák és kapcsolataik](#_4gow7rlxgmw6) 6

[1. users tábla](#_2v2dpcc954tv) 6

[2. words\_hu tábla](#_5fb0fvfl6v1h) 6

[Oldalak listája](#_f68eazj2k9j2) 7

[**A Szoftver fő funkciói**](#_9pmm3caa8lr9) **8**

[1. Játékmenet és vezérlők](#_n7i5epwng1tl) 8

[Felhasználói fiók kezelés](#_25tuut5sp26f) 8

[**Felhasználói dokumentáció**](#_io7392c7ludc) **9**

[Felhasználói fiók kezelése](#_9vo7v7p3d0a2) 9

[A játék indítása és menete](#_33uitv88miac) 10

[**Funkciók és megoldások**](#_ss3mifyftb9w) **11**

[Bábuk mozgatása](#_wmwc1hesdgoq) 11

[A játéktábla kialakítása](#_464r1fipqbu9) 12

[**A szoftver tesztelése**](#_f3n2gjplp4bs) **13**

[**További fejlesztések**](#_dvgp1qyvpl9h) **15**

[**Eredmények**](#_icv5r2odfkl0) **16**

[Funkcionális eredmények](#_gkvh1ym9ht1z) 16

[Technikai eredmények](#_wqk0wl9a6tsq) 16

[**Zárszó**](#_2bxgple1u5hn) **17**

# Bevezetés

Jelen dokumentáció az **Activity WEBition** projektet mutatja be, amely egy teljes mértékben webes alapon működő Activity társasjáték adaptáció. A projekt célja egy olyan digitális platform megteremtése volt, amely a klasszikus társasjáték élményét viszi át az online térbe, kiküszöbölve a fizikai játékokból adódó nehézségeket, mint az elveszett vagy sérült alkatrészek, illetve a hordozhatóság hiánya. A fejlesztés során a hangsúly a funkcionalitáson, a reszponzivitáson és a felhasználóbarát felület kialakításán volt.

A projekt egy kétfős csapatmunka részeként indult, ahol a társam a felhasználói fiókokért és az adminfelületért felelt volna, én pedig a játékmenetért. Sajnos a társam félúton feladta, így a projektet végül egyedül, a kezdeti tervektől eltérően, teljes egészében magam fejeztem be. Ez a kihívás lehetővé tette, hogy a projekt minden részletére koncentráljak, és teljes mértékben elsajátítsam mind a front-end, mind a back-end fejlesztési folyamatokat.

## Célcsoport

A játék elsődleges célcsoportja a családok és baráti társaságok, akik szeretik a társasjátékokat, és gyakran gyűlnek össze a szabadidejükben. A webes platformnak köszönhetően a játékhoz nincs szükség a fizikai dobozra és annak elemeire, így kényelmesen játszható bármilyen, internetkapcsolattal rendelkező eszközön. A felhasználói fiókok és a szóadatbázis bővítésének lehetősége miatt a játék különösen vonzó lehet azoknak, akik aktívan szeretnének részt venni a tartalom fejlesztésében és szeretnék személyre szabni a játékélményt. A reszponzív kialakítás révén a játék ideális utazások során, vagy ha a fizikai játék nem áll rendelkezésre.

## Projektcélok

A projekt fő célkitűzései a következők voltak:

* A fizikai Activity társasjáték digitalizálása, megőrizve az eredeti játékmenetet.
* Egy olyan webes alkalmazás létrehozása, amely minden modern eszközön (PC, tablet, mobiltelefon) zökkenőmentesen és reszponzívan fut.
* Egy megbízható adatbázis-kezelő rendszer kiépítése, amely lehetővé teszi a szavak tárolását és bővítését.
* Egy intuitív felhasználói felület kialakítása, amely egyszerűvé teszi a játék indítását és a játékmenetet a felhasználók számára.

# 

# Felhasznált technológiák

A projekt megvalósítása a webfejlesztés standard technológiáit és egy úgynevezett **LAMP stack** környezetet használ:

* **Frontend:**
  + **HTML5:** A felhasználói felület struktúrájának kialakítására.
  + **CSS3:** A vizuális megjelenítésért, a dizájn és a reszponzivitás megvalósításáért.
  + **JavaScript (jQuery):** A dinamikus viselkedésért, a bábu léptetéséért, az időzítőért és az AJAX kérések kezeléséért.
* **Backend:**
  + **PHP:** A szerveroldali logika, a felhasználói interakciók és az adatbázis-kommunikáció kezelésére.
* **Adatbázis:**
  + **MySQL:** Az adatok (szavak, felhasználói fiókok) tárolására és kezelésére.

# Rendszerkövetelmények

A projekt futtatásához egy **webkiszolgáló környezet** szükséges, amely támogatja a **PHP**-t és a **MySQL**-t. Helyi gépen a legegyszerűbb, ha egy előre konfigurált, csomagolt megoldást használsz, mint például a **WAMP**, **MAMP** vagy **XAMPP**. Ezek a szoftverek biztosítják az Apache webszervert, a PHP feldolgozó környezetet és a MySQL adatbázis-szervert, amik mind elengedhetetlenek a játék futtatásához.

## Minimális szoftverkövetelmények

* **PHP:** A kódok a modern PHP verziókkal (legalább 7.x) kompatibilisek, mivel a felhasználókezeléshez a biztonságos password\_hash funkciót használják.
* **MySQL:** Bármely stabil verzió megfelelő, a rendszer a standard SQL utasításokat használja az adatbázis-kapcsolathoz.
* **Webkiszolgáló:** Apache vagy Nginx.
* **Külső könyvtárak:** A játék a **jQuery** 3.7.1-es verzióját használja, amelyet a kód egy CDN-ről (Content Delivery Network) hív be. Ez a dinamikus elemek kezeléséhez és az **AJAX** kérések megvalósításához szükséges. A játékhoz **internetkapcsolat szükséges** a jQuery betöltéséhez, ha az nincs helyben tárolva.

### 

# Architektúra

Az alkalmazás egy tipikus **háromszintű webes architektúrát** követ, ami logikailag elkülöníti a bemutató, a logikai és az adatkezelő réteget.

* **Frontend (Kliensoldali):** Ez a réteg felel a felhasználói felület megjelenítéséért és a felhasználói interakciók gyűjtéséért. A **HTML** adja a struktúrát, a **CSS** a stílust, a **JavaScript** pedig a dinamikus viselkedést.
* **Backend (Szerveroldali):** A **PHP** szkriptek (pl. index.php, getword.php) alkotják. Ez a réteg dolgozza fel a kliensoldali kéréseket, kezeli a munkameneteket ($\_SESSION) és kommunikál az adatbázissal.
* **Adatbázis:** A **MySQL** tárolja az összes perzisztens adatot, mint a felhasználói fiókokat (users tábla) és a szavakat (words\_hu tábla).

# Használt szoftverek és funkcióik

* **Sublime Text:** Ez egy népszerű kódszerkesztő, ami a **HTML**, **CSS**, **PHP** és **JavaScript** kódok írására és szerkesztésére szolgált.
* **WAMP (vagy XAMPP):** Ez egy helyi webszerver csomag, ami az Apache, MySQL és PHP rendszerekkel biztosította, hogy a weboldalt és az adatbázis kapcsolatot **helyben, offline környezetben** lehessen tesztelni.
* **Chrome (és más böngészők):** A Chrome fejlesztői eszközei elengedhetetlenek voltak a kód **teszteléséhez és hibakereséséhez**. A különböző böngészőkben való tesztelés segít meggyőződni arról, hogy a weboldal **reszponzív** és minden platformon megfelelően jelenik meg.
* **Adobe Fireworks és Adobe Illustrator:** Ezek a grafikai programok a játékhoz használt vizuális elemek, mint a bábuk, a logó és a játéktábla grafikájának megtervezéséhez és szerkesztéséhez kellettek.
* **Total Commander:** Ezt a szoftvert **FTP kliensként** használtad, ami a helyi projektfájlok feltöltéséhez volt szükséges a webes tárhelyre. Ez a program lehetővé teszi a fájlok gyors és biztonságos átvitelét a fejlesztői környezetből az éles szerverre.
* **phpMyAdmin:** Ez egy böngészőből elérhető grafikus felület, ami megkönnyíti a **MySQL adatbázisok kezelését**. Segítségével egyszerűen lehetett létrehozni a táblákat (pl. a words\_hu és a users táblákat).
* **Git:** Egy verziókövető rendszer, ami a kódban végzett változtatások nyomon követésére és verziózására szolgál.

# Adatbázis

A projekt egy **MySQL** adatbázist használ, amely az összes állandó adatot tárolja. A struktúra egyszerű, a könnyű kezelhetőség és a gyors lekérdezések érdekében lett kialakítva. Két fő tábla található benne: a felhasználókat tároló users és a játék szavait tároló words\_hu. Az adatbázis megtervezésénél az **egyszerűség** és a **funkcionalitás** volt a fő szempont.

## Táblák és kapcsolataik

### 1. users tábla

Ez a tábla a felhasználói fiókokat kezeli. Az adatok alapján a felhasználók **regisztrálhatnak**, **bejelentkezhetnek** és a jelszavukat is megváltoztathatják. A jelszavak biztonságos tárolása érdekében a password mező **hashelve** van. A tábla e-mail alapú jelszó-visszaállítást is támogat.

* id (int, 11): Egyedi azonosító.
* username (varchar, 12): A felhasználó neve.
* email (varchar, 50): A felhasználó e-mail címe.
* password (varchar, 255): A hashelt jelszó.
* reset\_code (varchar, 32): A jelszó-visszaállításhoz használt ideiglenes kód.

### 2. words\_hu tábla

Ez a tábla a játék feladványait tárolja. Mivel a játék célja a bővíthetőség, a regisztrált felhasználók új szavakat adhatnak hozzá az adatbázishoz, így garantálva a folyamatosan megújuló tartalmat. A szavak a játékban használt különböző feladványtípusokat és nehézségi szintjeiket is tartalmazzák.

* id (bigint, 255): Egyedi azonosító.
* word (varchar, 30): Maga a feladvány.
* type (varchar, 1): A feladvány típusa ('r' - rajzolás, 'k' - körülírás, 'm' - mutogatás).
* point (int, 6): A feladványért kapható pontszám.
* added\_by (int, 11): Annak a felhasználónak az azonosítója, aki hozzáadta a szót.
* reported (tinyint, 1): Jelző, ha egy szót bejelentettek.
* deleted (tinyint, 1): Jelző, ha egy szót töröltek.

A két tábla között a kapcsolat a users.id és a words\_hu.added\_by mezőkön keresztül jön létre, ami lehetővé teszi, hogy nyomon kövesd, ki melyik szót adta hozzá.

Felépítés

A szoftverben az index.php főoldal dinamikusan jeleníti meg a következő oldalakat a URL-ben lévő paramétertől függően:

## Oldalak listája

* **Kezdőképernyő / Játék (game.php)**: Ez az alapértelmezett oldal, ha nincs más paraméter megadva a címsorban. Tartalmazza a játék indításához szükséges csapatválasztót, a játéktáblát, és a játékmenet vezérlőit, mint az időzítő és a kártyahúzó gomb.
* **Regisztráció (registration.php)**: Ez az oldal a felhasználók regisztrációját teszi lehetővé, ahol megadhatják felhasználónevüket, e-mail címüket és jelszavukat.
* **Beállítások (settings.php)**: A bejelentkezett felhasználók ezen az oldalon változtathatják meg a jelszavukat.
* **Új szó hozzáadása (addword.php)**: A bejelentkezett felhasználók ezen a felületen tudnak új szavakat és feladványokat hozzáadni az adatbázishoz.
* **Elfelejtett jelszó (forgotpassword.php)**: Ez az oldal segítséget nyújt a jelszó helyreállításához az e-mail cím megadásával.
* **Szabályok (rules.php)**: A kezdőképernyőről egy link vezet erre az oldalra, ahol a játékosok megtekinthetik a játék szabályait.

# 

# A Szoftver fő funkciói

A játék a következő kulcsfontosságú funkciókkal rendelkezik:

## 1. Játékmenet és vezérlők

* **Csapatválasztás és játék indítása:** A játék indítása előtt a felhasználók kiválaszthatják a csapatok színét, amelyek egyedi azonosítót kapnak a játék során. A **"Játék indítása"** gomb megnyomásakor a program ellenőrzi, hogy legalább két csapat van-e, és elindítja a játékot.
* **Feladványok:** A játéktábla mezőit a rendszer a feladvány típusai szerint osztja fel: rajzolás (r), körülírás (k) és mutogatás (m). A **"Új kártya"** gomb megnyomásával a JavaScript AJAX kérést küld a szervernek (getword.php), amely a bábu aktuális pozíciójának megfelelő típusú szót kér le az adatbázisból.
* **Időzítő és pontozás:** A **"Mehet"** gomb elindít egy 60 másodperces visszaszámlálót. A sikeres feladvány után a csapat a megszerzett pontszámmal (point) léptetheti a bábuját. A **"Bukta"** gomb jelzi, ha a feladvány nem sikerült, és a kör pontszerzés nélkül lezárul.

## Felhasználói fiók kezelés

* **Regisztráció:** A felhasználók egy egyszerű felületen regisztrálhatnak, megadva felhasználónevüket, e-mail címüket és egy jelszót. A regisztráció során a jelszavak **hashelve** tárolódnak, ami garantálja a biztonságot.
* **Bejelentkezés és kijelentkezés:** A felhasználók a regisztráció után be tudnak jelentkezni felhasználónévvel vagy e-mail címmel. A userAuth.php fájl végzi a hitelesítést. A **Menü** gomb segítségével a felhasználók ki is jelentkezhetnek.
* **Jelszókezelés:** A bejelentkezett felhasználók a **Beállítások** oldalon megváltoztathatják a jelszavukat. A rendszer támogatja a jelszó-visszaállítást is, amennyiben a felhasználó elfelejti a jelszavát.
* **Szavak hozzáadása:** A regisztrált felhasználók új szavakat adhatnak az adatbázishoz, meghatározva azok típusát és nehézségét, ezzel bővítve a játék tartalmát.

# 

# Felhasználói dokumentáció

Ez a dokumentáció végigvezet az **Activity WEBition** webes alkalmazás használatán. A szoftver célja, hogy a népszerű Activity társasjáték digitális változatát nyújtsa.

## Felhasználói fiók kezelése

A regisztrált felhasználók új szavakat adhatnak a játék adatbázisához, ezzel hozzájárulva a tartalom bővítéséhez.

**Regisztráció**

1. A kezdőképernyő jobb alsó sarkában található **"Menü"** gombra kattintva nyisd meg a felhasználói menüt.
2. Kattints a **"Regisztráció"** gombra, ami átvisz a regisztrációs oldalra.
3. Add meg a felhasználónevedet, e-mail címedet és a jelszavadat, majd kattints a **"Regisztráció"** gombra. A jelszavad biztonságosan, hashelve kerül tárolásra.

**Bejelentkezés és kijelentkezés**

* A menüben, a regisztrációs gomb alatt, a bejelentkezési űrlapon add meg a felhasználónevedet vagy e-mail címedet, és a jelszavadat. Kattints a **"Bejelentkezés"** gombra.
* Bejelentkezés után a menüben megjelenő **"Kijelentkezés"** gombbal tudsz kilépni a fiókodból.

**Jelszó módosítása és visszaállítása**

* Bejelentkezés után a menüben található **"Beállítások"** oldalon tudod megváltoztatni a jelszavadat.
* Ha elfelejtetted a jelszavadat, kattints a bejelentkezési űrlap alatt található **"Elfelejtett jelszó"** linkre. Add meg az e-mail címedet, és a rendszer elküld egy kódot a jelszó visszaállításához.

**Új szó hozzáadása**

* Bejelentkezés után a menüben megjelenik az **"Új szó hozzáadása"** gomb, amellyel a words\_hu adatbázist tudod bővíteni.
* Add meg a szót, válassz típust (rajzolás, körülírás, mutogatás), és adj meg egy pontszámot.

## A játék indítása és menete

1. **Csapatok kiválasztása**: A kezdőképernyőn a rendelkezésre álló figurák (piros, kék, sárga, zöld) közül választhatod ki a csapatokat. Egy csapathoz kattints a kívánt színű bábra, ami átkerül a "választott csapatok" részbe.
2. **Játék indítása**: Ha legalább két csapatot választottál, kattints a **"Játék indítása"** gombra.
3. **A játéktábla**: A tábla 49 mezőből áll. A mezők színe és ikonja jelzi, hogy milyen típusú feladványra számíthat az adott mezőn lévő csapat:
   * **Kék ceruza**: Rajzolás
   * **Zöld száj**: Körülírás
   * **Narancssárga pantomim figura**: Mutogatás
4. **A játékvezérlők**:
   * **Új kártya**: Kattints erre a gombra, hogy egy új szót kérj le az adatbázisból. A gomb színe jelzi, hogy melyik csapat következik.
   * **Mutat**: Tartsd lenyomva ezt a gombot a feladvány és a pontszám megtekintéséhez.
   * **Mehet**: Ezzel a gombbal indíthatod a 60 másodperces időzítőt, és kezdhetitek el a feladványt.
   * **Pontozó gombok**: Ha sikerült a feladvány, kattints a megfelelő pontszámot tartalmazó gombra. Ez lépteti a csapatod bábuját a táblán.
   * **Bukta**: Ha a feladvány nem sikerült, kattints erre a gombra, hogy befejezd a kört pontszerzés nélkül.
   * **Időzítő**: A képernyő tetején található, és visszaszámlálja a feladványra rendelkezésre álló időt. Ha lejár, hangjelzés és felirat figyelmeztet.
5. **Mobil nézet**: Kisebb kijelzőkön (pl. telefon) a játéktábla helyén csak a csapatok bábui jelennek meg, rajtuk az aktuális pozíciójukkal.

# Funkciók és megoldások

## Bábuk mozgatása

A bábuk mozgatásáért a **move(id)** nevű JavaScript függvény felel. A kód nem engedi, hogy a bábu azonnal a célmezőre ugorjon, hanem egy animált, lépésenkénti haladást valósít meg.

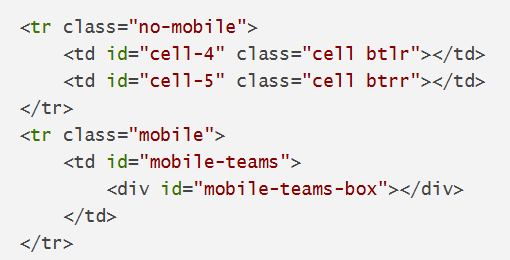
* **Célpozíció számítása:** Először a függvény kiszámítja a bábu új, végleges pozícióját (end) az aktuális pozíciójának és a megszerzett pontoknak az összeadásával.
* **Lépésenkénti animáció:** Ezután a kód a **setInterval()** függvényt használja, amely 200 ezredmásodpercenként futtatja a léptetés logikáját.
* **Vizuális frissítés:** Minden lépésnél a kód növeli a bábu logikai pozícióját (teamspos[id]++), majd a jQuery **append()** parancsával áthelyezi a bábu HTML-elemét ($('#' + teams[id])) a következő mezőbe ($('#cell-' + teamspos[id])). Ez a megoldás adja a mozgás érzetét.
* **Vége:** A folyamat addig folytatódik, amíg a bábu eléri a végpozíciót, ekkor a **clearInterval()** leállítja az animációt.



## A játéktábla kialakítása

A játéktábla kialakítása egy praktikus tervezési döntés eredménye.

* **A "huszárvágás":** Eleinte bonyolult PHP algoritmussal akartam legenerálni a tábla kígyózó útvonalát. Ezt a megközelítést végül elvetettem, mivel a manuális HTML kódolás sokkal gyorsabb és átláthatóbb megoldást eredményezett. A game.php fájl tartalmazza a tábla HTML struktúráját, ahol minden mező egyedi azonosítót kap a cell- prefixszel.
* **Kéttábla megjelenítés:** A játék reszponzív, és a CSS (style.css) segítségével két különböző nézetet kezel.
  + **Asztali nézet:** Nagyobb képernyőkön a táblát a **.no-mobile** osztály teszi láthatóvá.
  + **Mobil nézet:** Kisebb kijelzőkön a **.mobile** osztály jelenik meg, ami elrejti a teljes játéktáblát, és helyette csak a csapatok bábuit és pontszámait mutatja meg egy egyszerűsített listában. Ez a megoldás a játékot játszhatóvá teszi kisebb kijelzőkön is.



### 

# A szoftver tesztelése

A tesztelés célja a szoftver funkcionalitásának, megbízhatóságának és felhasználóbarát jellegének ellenőrzése. A tesztek lefuttatása biztosítja, hogy a program a felhasználói elvárásoknak megfelelően viselkedjen, és a felmerülő hibákat még a kiadás előtt orvosolni lehessen. Az alábbi táblázatban bemutatott tesztek a legkritikusabb funkciókra fókuszálnak, mint a regisztráció, a játékmenet és a bábu mozgatása.

| **Teszt azonosítója** | **Teszt rövid leírása** | **Hibajelenség (ha van)** | **Javítás** | **Javítás sikeressége** |
| --- | --- | --- | --- | --- |
| TEST-001 | A regisztrációs űrlap kitöltése érvényes felhasználónévvel, e-mail címmel és jelszóval. | Nincs | Nincs | Sikeres |
| TEST-002 | Próbálkozás regisztrációval egy már létező felhasználónévvel vagy e-mail címmel. | A rendszer hibát jelez az e-mail vagy a felhasználónév lefoglaltsága miatt. | Nincs, a hibaüzenet megfelelő. | Sikeres |
| TEST-003 | Egyetlen csapat kiválasztása, majd a "Játék indítása" gombra kattintás. | A játék elindul, noha minimum 2 csapatra van szükség a szabályok szerint. A rendszer nem ad hibaüzenetet. | JavaScript kódot kell módosítani, hogy a start() függvényben ellenőrizze a csapatok számát. Ha a szám kevesebb, mint 2, egy figyelmeztető üzenet jelenik meg (alertBox). | Sikeres |
| TEST-004 | Egy 6 pontos feladvány sikeres kitalálása és a pontozó gomb megnyomása. | A bábu nem megfelelő számú mezőt lép, vagy a játék hibát jelez a 49-es mező elérésekor. | A move() függvény logikáját kell ellenőrizni. Meggyőződni arról, hogy a point változó helyesen kerül átadásra a függvénynek, és a célmező (49) elérésekor a kód a gratula üzenettel leállítja a játékot. | Sikeres |
| TEST-005 | Reszponzivitás ellenőrzése, a weboldal megnyitása mobilon vagy kisebb kijelzőn. | A játéktábla nem tűnik el, vagy a bábuk és a pontszámok megjelenítése hibásan jelenik meg. | A CSS-ben a @media only screen and (max-width: 900px) szabályt kell ellenőrizni, és megbizonyosodni arról, hogy a .no-mobile elemek eltűnnek, a .mobile elemek pedig megjelennek. | Sikeres |
| TEST-006 | AJAX kérés hibakezelése, a getword.php fájlhoz tartozó AJAX kérés elindítása hálózati hiba esetén (pl. megszakított internetkapcsolat). | A játék megakad, vagy nem ad felhasználóbarát hibaüzenetet a probléma okáról. | A $.ajax() kéréshez hozzá kell adni egy hibakezelő (.fail()) függvényt, amely hibaüzenetet jelenít meg a felhasználó számára. | Kisebb hibák maradtak, de alapvetően sikeres |

# További fejlesztések

A fejlesztések itt nem álltak meg, hiszen a jövőben számos új funkcióval és bővítéssel tehetjük még jobbá a játékot. Ezek a tervek a játékélményt és a felhasználói közösséget is tovább gazdagítják.

* **Szavak moderálása:** Egy moderációs rendszer bevezetése a felhasználók által hozzáadott szavak minőségének biztosítására. Ez megakadályozná a nem megfelelő vagy értelmetlen bejegyzéseket, és fenntartaná a szótár tisztaságát.
* **Beépített rajztábla:** A "rajzolás" típusú feladványokhoz egy digitális rajzfelület beépítése lehetővé tenné a játékosok számára, hogy közvetlenül az eszközükön rajzoljanak, így nincs szükség tollra és papírra.
* **Játéksession mentése:** A játékállapot adatbázisba való mentése lehetővé tenné a játékosok számára, hogy később folytassák a félbehagyott játékot, ami nagyobb rugalmasságot és kényelmet biztosít.
* **Online multiplayer:** A játék kiterjesztése, hogy a játékosok távolról is versenyezhessenek, egy új dimenziót nyitna, és lehetővé tenné, hogy a barátok a helytől függetlenül együtt játszhassanak.
* **Többnyelvűség:** A szavak és a felhasználói felület több nyelven történő elérhetővé tétele szélesebb közönség számára tenné a játékot hozzáférhetővé.
* **Változtatható játékszabályok:** A játékszabályok testreszabásának lehetősége, például az időkorlátok vagy a pontértékek beállítása, változatosabb és személyre szabottabb játékmenetet tenne lehetővé.

### 

# Eredmények

A projekt sikeresen teljesítette a kitűzött célokat, és egy teljes mértékben funkcionális, digitális Activity társasjátékot eredményezett. A kezdeti kihívások és a társam kiszállása ellenére a projekt minden fontos eleme elkészült és kifogástalanul működik.

## Funkcionális eredmények

* **Teljes körű játékmenet**: A szoftver képes a teljes játékmenet kezelésére, a csapatválasztástól a játéktáblán való lépegetésig.
* **Feladványok és időzítő**: A játék sikeresen lekérdezi és megjeleníti a szavakat a megfelelő pontszámmal. A beépített időzítő pontosan visszaszámlál és jelzi a kör végét.
* **Felhasználói fiókok**: A regisztráció, bejelentkezés és kijelentkezés biztonságos, a jelszavak hashelve tárolódnak az adatbázisban. A felhasználók hozzáadhatnak új szavakat, így bővítve a játék tartalmát.

## Technikai eredmények

* **Reszponzív kialakítás**: A játékfelület minden modern eszközön, beleértve a telefonokat és a tableteket is, reszponzívan jelenik meg. A mobil nézetben a játéktábla helyén az egyszerűsített pontszámkijelző jelenik meg, ami optimalizált játékélményt biztosít a kisebb képernyőkön is.
* **Sikeres bábu léptetés**: A bábu animált mozgása, ami egy technikai kihívást jelentett, sikeresen megvalósult. A **setInterval** JavaScript függvény lépésről lépésre halad a táblán, vizuálisan követhetővé téve a játék menetét.
* **Robusztus háttérrendszer**: A PHP-alapú háttér sikeresen kezeli az **AJAX** kéréseket, lehetővé téve a szavak betöltését az oldal újratöltése nélkül. Az adatbázis-kapcsolat stabil és megbízható.

# Zárszó

Az **Activity WEBition** projekt sikeresen teljesítette a kitűzött célt: egy teljesen funkcionális, reszponzív webes játékot hoztam létre, amely hű marad az eredeti társasjáték szellemiségéhez. Bár a projekt a kezdeti csapattervektől eltérően végül egyedül készült el, ez a kihívás lehetővé tette, hogy a fejlesztési folyamat minden részletét mélyebben megismerjem, a front-end dizájntól a back-end logikáig.

A projekt nem csupán a technikai tudásomat bővítette, hanem a problémamegoldó képességemet és a kitartásomat is fejlesztette. Az elért eredmények, mint az animált bábu mozgatása, a reszponzív felület, és a biztonságos felhasználói rendszer, mind a gondos tervezés és a kitartó munka gyümölcsei. A szoftver egy stabil alap, amely a jövőbeni fejlesztésekkel (online multiplayer, beépített rajztábla) tovább bővíthető és fejleszthető.

**Köszönöm a figyelmet!**