Adatbázis alapú rendszerek

2023-2024/2

# IB152L-10

**Quiz**

**Készítette:**

Kávai Illés

Kaszás Zsolt József

Velicán-Benkő Attila

## Munka felosztása

Közösen dolgozunk a feladatok megoldásán.

## Értékelési mód:

A csoport tagjai közösen dolgoznak, minden tag ugyanannyi pontot kap.

## Feladat szöveges leírása

A weboldal egy quiz játék. Amelyben a felhasználók saját szobát készíthetnek. Beállíthatnak saját kérdéseket azokra válaszokat. Ezekhet a szobákhoz más játékosok csatlakozhatnak és megnézhetik, hogy mennyi kérdésre tudnak válaszolni az adott szobában.

A weboldal, php-ban készül. Az Oracle 12.0.0.0.1 adatbázist és a SQLDevelopert használjuk az adatbázis létrehozásához és kezeléséhez. A dokumentációhoz a word programot választottuk. Az alkalmazás fejlesztése Visual studio code/ Webstorm fejlesztői környezetben történik. Diagrammok készítésére a következőket használjuk: Draw.io, ClickCharts.

Az adatbázis tartalmazza az hét fő adattáblát: felhasznalo, eredmeny, kerdes, tema, valasz, *szoba,* szoba\_kerdesei. Az adattáblákat az adatbázis kezeli és frissíti az alkalmazás által.

Az alkalmazás a belépő képernyővel indul, ahol a felhasználó beléphet a felhasználói fiókjába. Ha a felhasználó még nem regisztrált, akkor a regisztrációs űrlapot kell kitöltenie. A sikeres bejelentkezés után a felhasználó a főmenübe kerül, ahol lehetőség van, szobát létrehozni, szobába csatlakozni, kijelentkezni, profil megjelenitésre.

A szobába való belépést követően lehetőség van kitölteni a quizt. A játékban mindig csak egy kérdés aktív, és a válaszok mindig csak a kérdésre vonatkoznak.

A játékmenet során a játékosok a quiz kitöltőivel versenyeznek, eredményük eltárolásra kerül az adatbázisban. A játék során a játékosok különböző kérdésekre válaszolnak. Az adott kérdésekhez tartozó válaszokat az adattáblákból olvassa be a program, és a válaszok helyességét is az adatbázisban tárolja.

## Követelménykatalógus

Cél: Az alkalmazás célja, hogy interaktív és szórakoztató módon lehetőséget biztosítson a felhasználóknak szobákban vagy nyilvános helyeken való versengésre, ahol különböző témákhoz kapcsolódó kérdésekre válaszolhatnak.

Funkcionális követelmények:

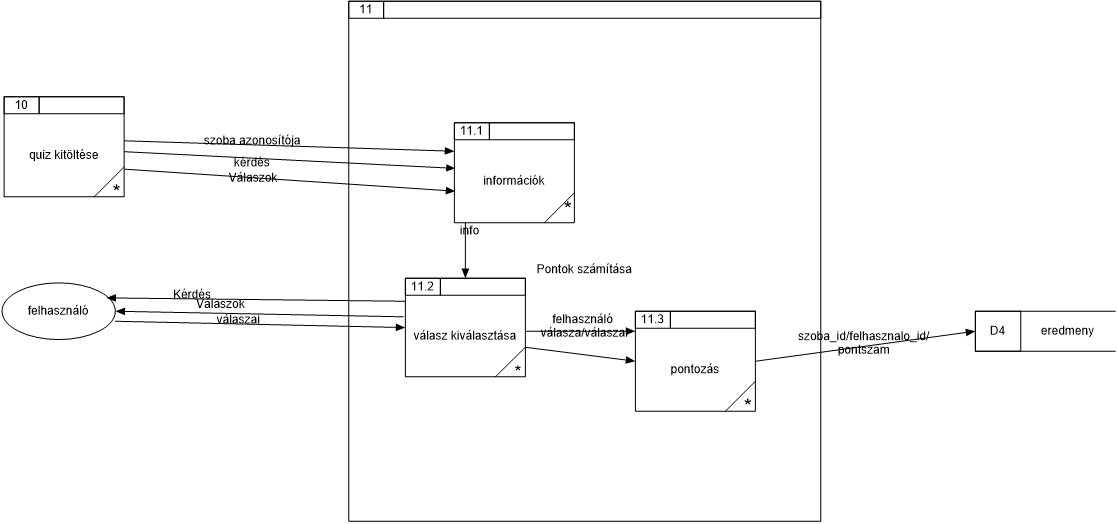
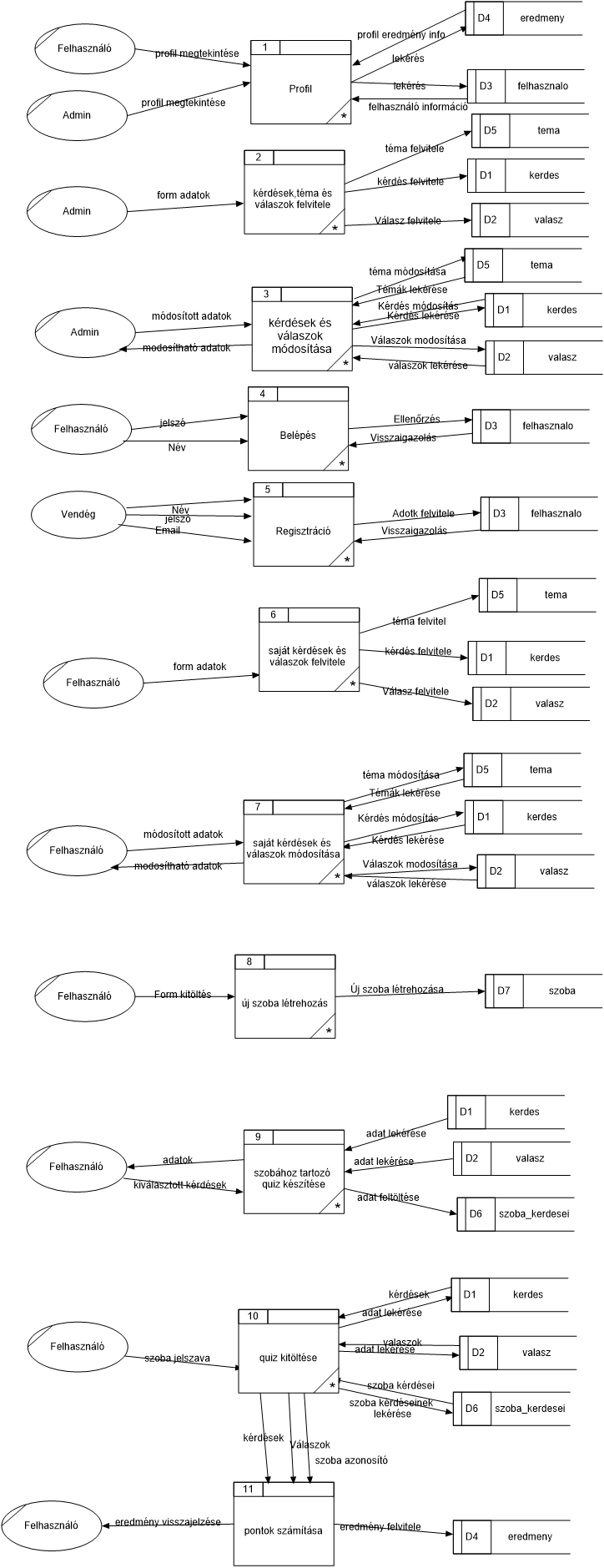
* A játékosoknak lehetősége kell, hogy legyen a alkalmazásba való belépésre egy felhasználónév és jelszó segítségével.
* Regisztráció
* A játékosoknak lehetősége kell, hogy legyen a játékszobákhoz való csatlakozásra
* A játékosoknak lehetősége kell, hogy legyen új játékszobák létrehozására, amelyeknek nevét és jelszavát meg kell adniuk.
* A játékosoknak lehetősége kell, hogy legyen a kérdésekre való válaszadásra, amelyek véletlenszerűen kerülnek megjelenítésre.
* A játékosok pontokat kell, hogy szerezzenek a helyes válaszokért, és az adott időn belül kell válaszolniuk a kérdésekre.

Nem funkcionális követelmények:

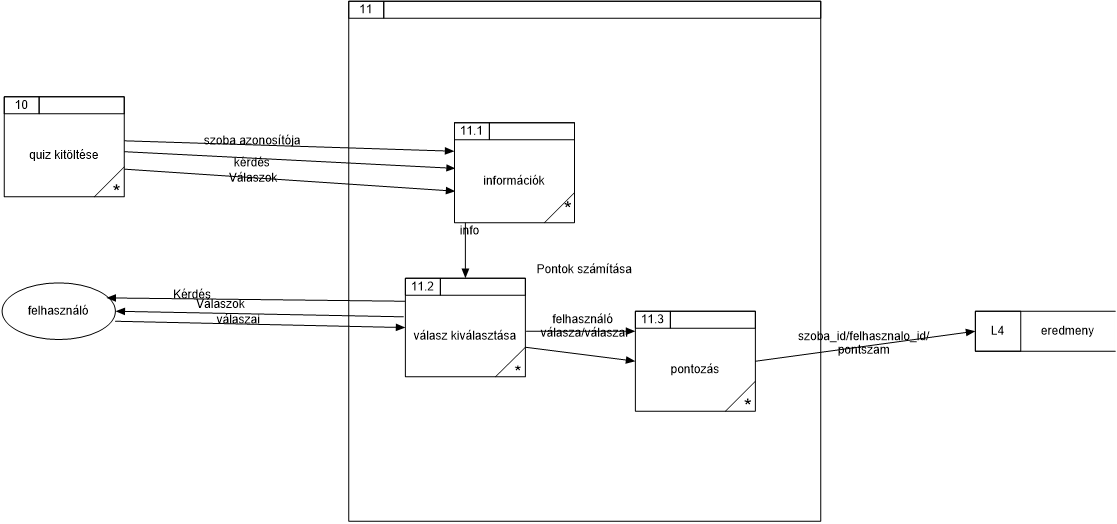
* Megbízhatóság
* Teljesítmény
* Felhasználói felület
* A programnak grafikus felhasználói felületet kell biztosítania a felhasználók számára.
* A felhasználói felületnek ergonomikusnak kell lennie, és az információkat jól átlátható módon kell megjelenítenie.
* A felhasználói felületnek tartalmaznia kell a szükséges gombokat és elemeket, amelyekkel a felhasználók könnyen használhatják a programot.
* Karbantarthatóság
* Biztonság
* A programnak biztosítania kell az adatok biztonságos tárolását és kezelését.
* A programnak lehetővé kell tennie a felhasználók jogosultságainak kezelését, például az adminisztrátori jogosultságokat.
* A programnak megfelelő védelmet kell biztosítania a bizalmas adatokkal szemben, például a jelszavak védelmével.

## Adatfolyam diagram (DFD):

Fizikai



Logikai



## 

## Egyedmodell:



## EKT-diagram:

## 

**Egyed-kapcsolat diagram leképezése relációs adatbázissémákká:**

felhasznalo (id, nev, email, jelszo, admin-e)

eredmeny (id, *szoba\_id*, *felhasznalo\_id*, pontszam)

kerdes (id, kerdes, *tema\_id*)

tema (id, nev)

valasz (id, *kerdes\_id*, valasz ,helyes\_e)

szoba (id, nev, jelszo, *felhasznalo\_id*)

szoba\_kerdesei(id, *szoba\_id*, *kerdes\_id*)

eredmenyei(*felhasznalo.id* , *eredmeny.felhasznalo\_id)*

szobája(*felhasznalo.id* , *szoba.felhasznalo\_id*)

kerdesei(*kerdes.id* , *szoba\_kerdesei. kerdes\_id*)

## 

## Relációs adatelemzés

1. normálforma (1NF):

* felhasznalo: (id, nev, email, jelszo, admin-e)
* eredmeny: (id, szoba\_id, felhasznalo\_id, pontszam)
* kerdes: (id, kerdes, tema\_id)
* tema: (id, nev)
* valasz: (id, kerdes\_id, valasz ,helyes\_e)
* *szoba*: (id, nev, jelszo, felhasznalo\_id)
* szoba\_kerdesei: (id, szoba\_id, kerdes\_id)

Megjegyzés: Az adatbázis már eleve kiindulópontjában megfelel az 1NF-nek.

2. normálforma (2NF):

* felhasznalo: (id, nev, email, jelszo, admin-e)
* eredmeny: (id, *szoba\_id*, *felhasznalo\_id*, pontszam)
* kerdes: (id, kerdes, *tema\_id*)
* tema: (id, nev)
* valasz: (id, *kerdes\_id*, valasz ,helyes\_e)
* szoba: (id, nev, jelszo, *felhasznalo\_id*)
* szoba\_kerdesei: (id, *szoba\_id*, *kerdes\_id*)

Megjegyzés: Az adatbázis már eleve kiindulópontjában megfelel az 2NF-nek, hiszen minden táblának van egyetlen elsődleges kulcsa, és nincsenek benne összetett attribútumok.

3. normálforma (3NF):

* felhasznalo: (id, nev, email, jelszo, admin-e)
* eredmeny: (id, *szoba\_id*, *felhasznalo\_id*, pontszam)
* kerdes: (id, kerdes, *tema\_id*)
* tema: (id, nev)
* valasz: (id, *kerdes\_id*, valasz ,helyes\_e)
* szoba: (id, nev, jelszo, *felhasznalo\_id*)
* szoba\_kerdesei: (id, *szoba\_id*, *kerdes\_id*)

Megjegyzés: Az adatbázis már eleve kiindulópontjában megfelel az 3NF-nek, hiszen az egyes táblákban található attribútumok között nincs transzitív függőség.

## Táblák leírása:

felhasznalo: *felhasznalo tartalmának leírása.*

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Leírás** |
| id | NUMBER | egyedi azonosító |
| nev | CHAR | felhasználó belépési neve |
| email | CHAR | Email- cím |
| jelszo | CHAR | jelszó |
| admin-e | INTEGER | adminisztrátor-e vagy sem |

eredmeny: *Ez az adattábla tárolja a játékos eredményét egy adott szobában*

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Leírás** |
| id | NUMBER | egyedi azonosító |
| szoba\_id | FOREIGN KEY | Szoba azonosítója |
| felhasznalo\_id | FOREIGN KEY | Felhasználó azonosítója |
| pontszam | INTEGER | pontszámuk |

kerdes: *Ez az adattábla tárolja az összes kérdést, amelyek az alkalmazásban megjelennek.*

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Leírás** |
| id | NUMBER | egyedi azonosító |
| kerdes | CHAR | kérdés szövege |
| tema\_id | FOREIGN KEY | Téma azonosítója |

tema: *Ez az adattábla tárolja az összes témát, amelyek az alkalmazásban megjelennek.*

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Leírás** |
| id | NUMBER | egyedi azonosító |
| nev | CHAR | Téma megnevezése |

valasz : *Ez az adattábla tartalmazza az összes választ, amelyeket a felhasználók az egyes kérdésekre adhatnak*

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Leírás** |
| id | NUMBER | egyedi azonosító |
| kerdes\_id | FOREIGN KEY | válaszhoz tartozó kérdés azonosítója |
| valasz | CHAR | válasz szövege |
| helyes\_e | INTEGER | A kérdése vonatkozó helyesség |

*szoba: Ez az adattábla tartalmazza a szoba információját*

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Leírás** |
| id | NUMBER | egyedi azonosító |
| nev | CHAR | szoba neve |
| jelszo | CHAR | jelszó |
| felhasznalo\_id | FOREIGN KEY | Azon felhasználó azonosítója, aki a szobát létrehozta |

szoba\_kerdesei : *Ez az adattábla tartalmazza az adott szobához tartozó kérdéseket*

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Leírás** |
| id | NUMBER | egyedi azonosító |
| szoba\_id | FOREIGN KEY | Szoba azonosítója |
| kerdes\_id | FOREIGN KEY | válaszhoz tartozó kérdés azonosítója |

## Szerep-funkció mátrix:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | Funkciók /Szerepkörök | Felhasználó | Admin | Vendég | | Regisztráció |  |  | x | | Belépés | x | x |  | | Profil | x | x |  | | saját kérdések és válaszok felvitele | x | x |  | | saját kérdések és válaszok módosítása | x | x |  | | új szoba létrehozás | x | x |  | | szobához tartozó quiz készítése | x | x |  | | quiz kitöltése | x | x |  | | kérdések és válaszok módosítása |  | x |  | | kérdések,téma és válaszok felvitele |  | x |  | |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

Hozzáadott triggerek:

Trigger: update\_last\_activity\_trigger

Leírás:

Ez a trigger minden alkalommal frissíti a felhasznalo tábla utolso\_aktivitas\_datum mezőjét, amikor egy új rekordot szúrnak be az eredmeny táblába, és az új rekordnak van egy felhasznalo\_id értéke, amely megegyezik a felhasznalo tábla egyik rekordjának azonosítójával.

Kód:

CREATE OR REPLACE TRIGGER update\_last\_activity\_trigger

BEFORE INSERT ON eredmeny

FOR EACH ROW

BEGIN

UPDATE felhasznalo

SET utolso\_aktivitas\_datum = CURRENT\_TIMESTAMP

WHERE id = :new.felhasznalo\_id;

END;

/

Trigger: update\_room\_last\_activity\_trigger

Leírás:

Ez a trigger minden alkalommal frissíti a szoba tábla utolso\_aktivitas\_datum mezőjét, amikor új rekordot szúrnak be, meglévő rekordot frissítenek vagy rekordot törölnek a szoba\_kerdesei táblában, és az adott rekordokhoz tartozó szoba\_id érték alapján azonosítja a megfelelő szobát.

Kód:

CREATE OR REPLACE TRIGGER update\_room\_last\_activity\_trigger

AFTER INSERT OR UPDATE OR DELETE ON szoba\_kerdesei

FOR EACH ROW

DECLARE

v\_room\_id szoba.id%TYPE;

BEGIN

-- Szoba azonosítójának lekérdezése az érintett sor alapján

IF INSERTING THEN

v\_room\_id := :new.szoba\_id;

ELSIF UPDATING THEN

v\_room\_id := :new.szoba\_id;

ELSIF DELETING THEN

v\_room\_id := :old.szoba\_id;

END IF;

-- Szoba utolsó aktivitás dátumának frissítése

UPDATE szoba

SET utolso\_aktivitas\_datum = CURRENT\_TIMESTAMP

WHERE id = v\_room\_id;

END;

/

Hozzáadott eljárások:

Eljárás: get\_average\_score

Leírás:

Ez az eljárás egy felhasználó által elért átlagos pontszám lekérdezését végzi a megadott felhasználó azonosítója alapján. Az eredményt az avg\_score kimeneti paraméterben adja vissza.

Kód:

CREATE OR REPLACE PROCEDURE get\_average\_score(user\_id IN NUMBER,avg\_score OUT NUMBER)

IS

BEGIN

SELECT AVG(pontszam)

INTO avg\_score

FROM eredmeny

WHERE felhasznalo\_id = user\_id;

END;

/

Eljárás: get\_all\_room\_results

Leírás:

Ez az eljárás az összes szobához tartozó eredményeket kérdezi le.

Kód:

CREATE OR REPLACE PROCEDURE get\_all\_room\_results (

result OUT SYS\_REFCURSOR

)

IS

BEGIN

OPEN result FOR

SELECT e.szoba\_id, e.felhasznalo\_id, e.pontszam

FROM eredmeny e;

END;

/

Eljárás: get\_toplist\_for\_room

Leírás:

Ez az eljárás az adott szobához tartozó legjobb eredményeket kéri le. A felhasználó nevét és a pontszámot adja vissza egy kimeneti kurzoron keresztül, rendezve a pontszámok csökkenő sorrendjében.

Kód:

CREATE OR REPLACE PROCEDURE get\_toplist\_for\_room (

room\_id IN NUMBER,

result OUT SYS\_REFCURSOR

)

IS

BEGIN

OPEN result FOR

SELECT f.nev AS felhasznalonev, e.pontszam

FROM eredmeny e

JOIN felhasznalo f ON e.felhasznalo\_id = f.id

WHERE e.szoba\_id = room\_id

ORDER BY e.pontszam DESC;

END;

/

Összetett lekérdezések:

Quiz.sql 137.sor -142.sor

SELECT f.nev AS felhasznalonev, e.pontszam

        FROM eredmeny e

        JOIN felhasznalo f ON e.felhasznalo\_id = f.id

        WHERE e.szoba\_id = room\_id

        ORDER BY e.pontszam DESC;

quiz\_toplist.php 40.sor - 44.sor

SELECT s.nev AS szoba\_neve

    FROM szoba s

    JOIN eredmeny e ON s.id = e.szoba\_id

    WHERE e.felhasznalo\_id = :user\_id AND s.id = :room\_id

    ORDER BY e.id DESC

new\_question.php 77.sor

SELECT k.id AS question\_id, k.kerdes AS question\_text, v.id AS answer\_id, v.valasz AS answer\_text, v.helyes\_e AS is\_correct, k.globalis\_kerdes AS is\_global FROM kerdes k LEFT JOIN valasz v ON k.id = v.kerdes\_id

quiz.php 85.sor - 89.sor

SELECT k.\*, v.\*

        FROM szoba\_kerdesei sk

        JOIN kerdes k ON sk.kerdes\_id = k.id

        JOIN valasz v ON k.id = v.kerdes\_id

        WHERE sk.szoba\_id = :room\_id