

KAPPA CSAPAT

Webtárhely, VPS, Levelező szolgáltatás-regisztráció

Csapattagok:

- Csupity Tamás - AGLU8T (Csapatvezető)
 - Gila Ferenc - GVPXRN
 - Mihály Marcell - FKNKDH

ADATBÁZIS ALAPÚ RENDSZEREK GYAKORLATA (IB152I)

Szerda: 14:00-16:00

2025. tavaszi szemeszter

A dokumentációt: Csupity Tamás készítette

Specifikáció:

Cél:

A projekt célja egy webalapú alkalmazás fejlesztése, amely lehetővé teszi a webtárhely, VPS, levelezési szolgáltatások és domainek egyszerű és hatékony kezelését mind az ügyfelek, mind az adminisztrátorok számára. A rendszer automatizáltan biztosítja az számlázást és statisztikai elemzéseket.

A felhasználó regisztrálni tud majd bejelentkezni, elő tud fizetni díjcsomagokra amelyek domain és webszerver szolgáltatások lehetnek. Megtudja nézni a gyakorta ismételt kérdéseket és az üzemeltetők válaszait. A rendszer számlázni fog az ügyfélnek aki ezeket ki tudja egyenlíteni. különböző érdekes statisztikákat tud megtekinteni amelyek a szolgáltató vagy a szolgáltatásokkal kapcsolatosak. Az admin felhasználót, számlát, díjcsomagot, tudástárat, statisztikát, domiant és webtárhelyet tud módosítani, hozzáadni és törölni.

Funkcionális követelmények

Alaprendszer funkciói:

- Webtárhelyek, adatbázisok kezelése
- Felhasználókezelés: Regisztráció, bejelentkezés, felhasználói profilok kezelése, jogosultságok beállítása.
- Domainek kezelése
- Webtárhely kezelése
- Díjcsomagok (különböző tárhely- és adatbázis-kapacitás)
- Számlakészítés.

Bővített funkciók:

- Tudástár (gyakran ismételt kérdések és válaszai a webhostinggal kapcsolatban)
- Díjcsomagok ajánlása a felhasználónak havi szinten a forgalom, felhasznált tárhely szerint
- Számlák mentése havonta, számlák állapotváltozásainak kezelése az adatbázisba
- Bevételi statisztika, kimutatás készítés
- Korlátozás, esetleges kitiltás fizetési késedelem esetén
- Legnézettebb oldalak kiszűrése
- Legaktívabb felhasználók kiszűrése
- Legtöbbet fizető tulajdonosok kiszűrése

Feladatok kiosztása az 1. mérföldkőben:

Gila Ferenc - GVPXRN: (33,3%)

- Egyedmodell
- Egyed-esemény mátrix
- Szerep-funkció mátrix

Mihály Marcell - FKNKDH: (33,3%)

- Logikai adatfolyam-diagramok
- Fizikai adatfolyam-diagramok

Csupity Tamás - AGLU8T: (33,3%)

- Egyed-kapcsolat diagram
- Egyed-kapcsolat diagram leképezése relációs adatbázissémákká
- Funkcionális függőségek felírása, relációsémák normalizálása 3NF-ig

Mindenki:

- Specifikáció, részletes feladtleírás, követelménykatalógus
- Feladat felosztás és dokumentum készítés

Feladatok kiosztása 2. mérföldkőben:

Gila Ferenc - GVPXRN: (33,3%)

Mihály Marcell - FKNKDH: (33,3%)

Csupity Tamás - AGLU8T: (33,3%)

- A DDL parancsok létrehozása: Mindenki
- A DML parancsok létrehozása: Mindenki
- ON DELETE , kulcsok egyéb feltételek hozzáadása: Mindenki
- Feltöltés adatokkal: Mindenki

Feladatok kiosztása 3. mérföldkőben:

- Bejelentkezés és regisztráció megvalósítása, admin felvitel: (5p)

Mihály Marcell - FKNKDH: (50%)

Csupity Tamás - AGLU8T: (50%)

- Alapadatokat tartalmazó táblákhoz adatfelvitel, módosítás és törlés megvalósítása úrlapon keresztül: (0–1 pont (legfeljebb 12 tábláig))

Gila Ferenc - GVPXRN: 1 pont

A maradék fennmaradó ponton:

Mihály Marcell - FKNKDH: (50%)

Csupity Tamás - AGLU8T: (50%)

Feladatok kiosztása 4. mérföldkőben:

- Alapadatokat tartalmazó táblák adataihoz lekérdezések készítése (kötelező) – értékelés: táblánként 0–1 pont (legfeljebb 12 tábláig)

Gila Ferenc - GVPXRN: 0 pont

A maradék fennmaradó ponton:

Mihály Marcell - FKNKDH: 30%

Csupity Tamás - AGLU8T: 70%

- Triggerek írása a specifikációban jelölt funkciókhoz (kötelező*) – értékelés: triggerenként 0–3 pont

Gila Ferenc - GVPXRN: 2 pont

A maradék fennmaradó ponton:

Mihály Marcell - FKNKDH: 10 %

Csupity Tamás - AGLU8T: 90%

- Tárolt eljárások/függvények a megjelölt funkciókhoz (kötelező**) – értékelés: darabonként 0–3 pont

Gila Ferenc - GVPXRN: 2 pont

A maradék fennmaradó ponton:

Mihály Marcell - FKNKDH: 20%

Csupity Tamás - AGLU8T: 80%

- Funkciókat megvalósító összetett lekérdezések (kötelező***) – értékelés: lekérdezésenként 0–2 pont

Gila Ferenc - GVPXRN: 0

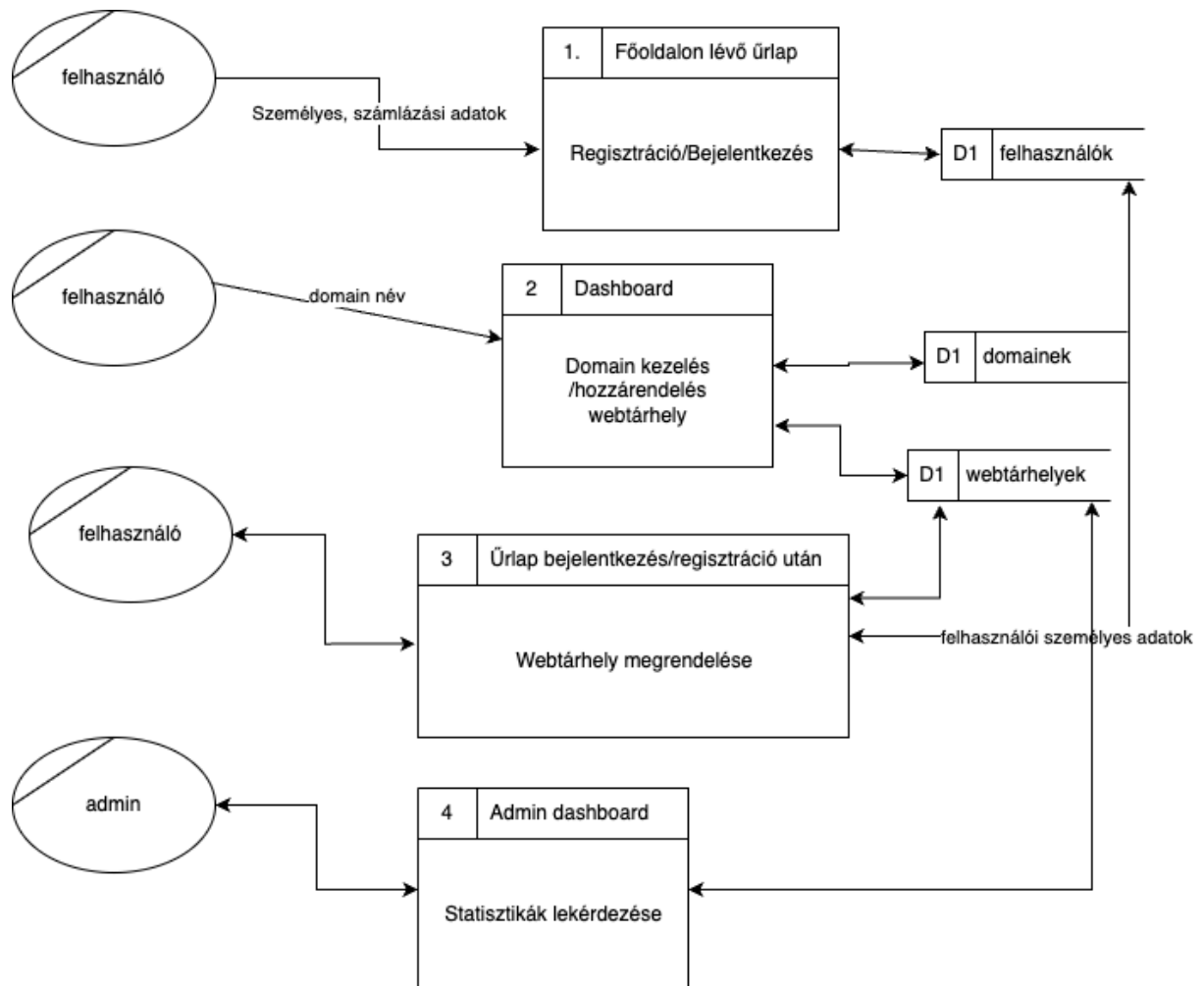
A maradék fennmaradó ponton:

Mihály Marcell - FKNKDH: 20%

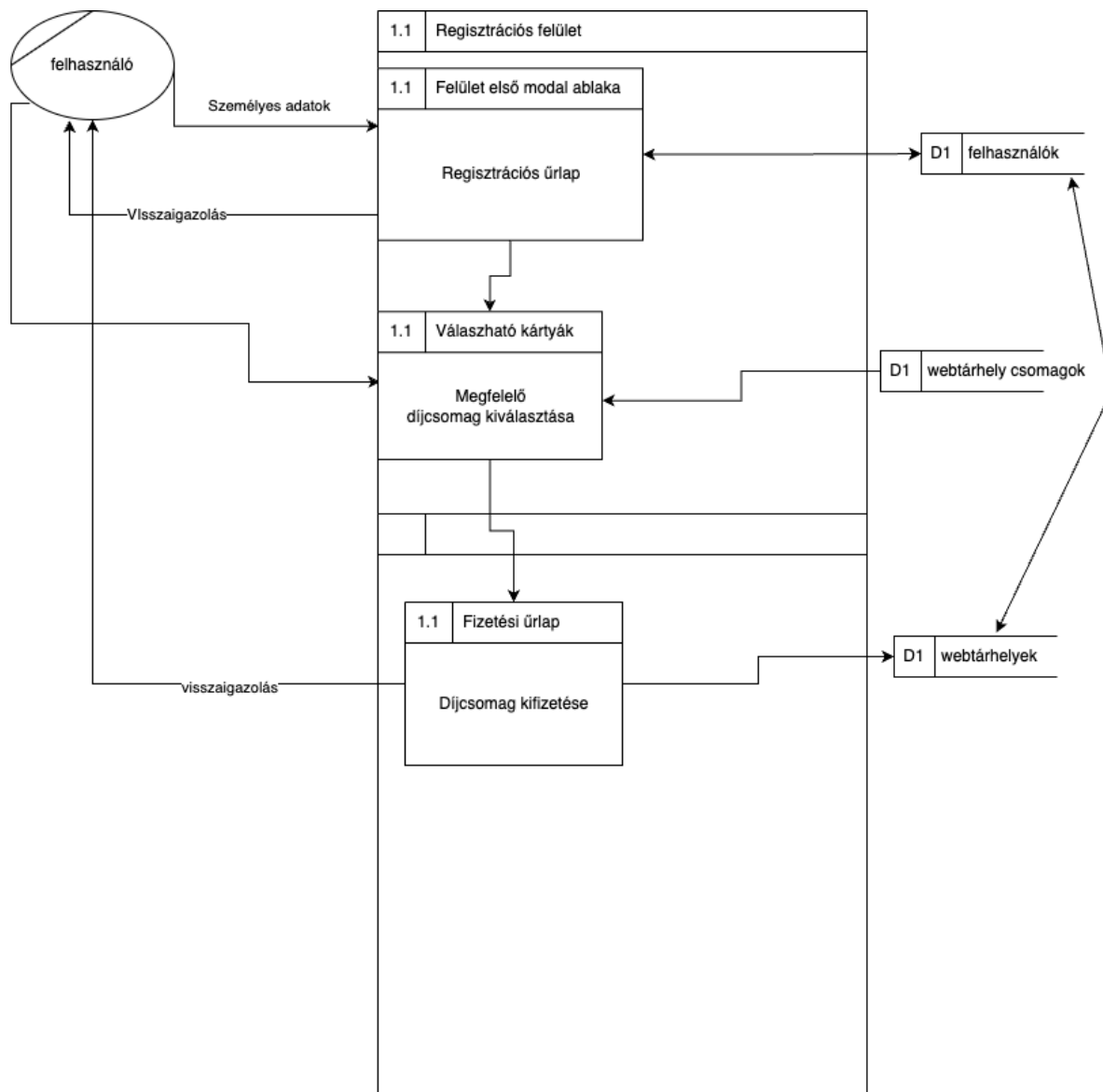
Csupity Tamás - AGLU8T: 80%

Tervezés, SSADM

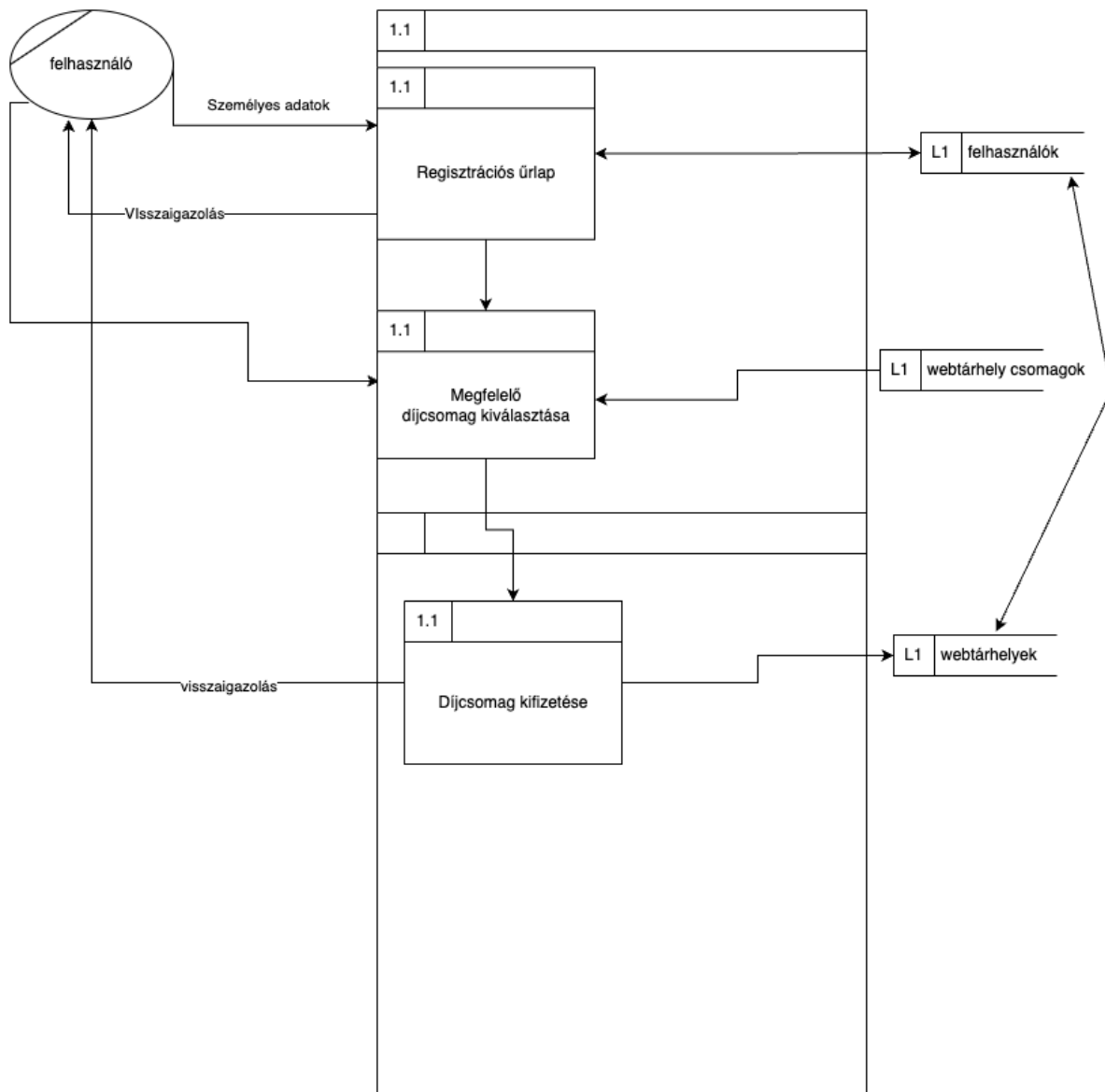
Fizikai adatfolyam:



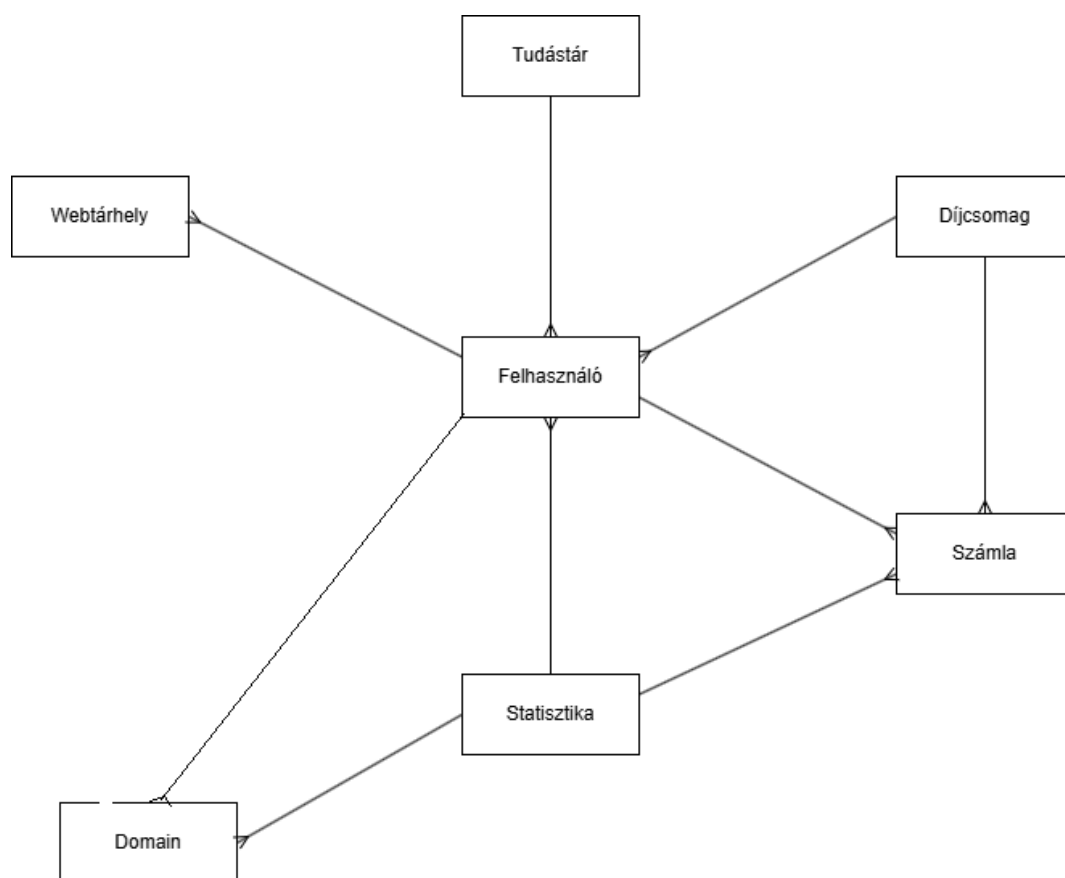
Logikai adatfolyam 1.:



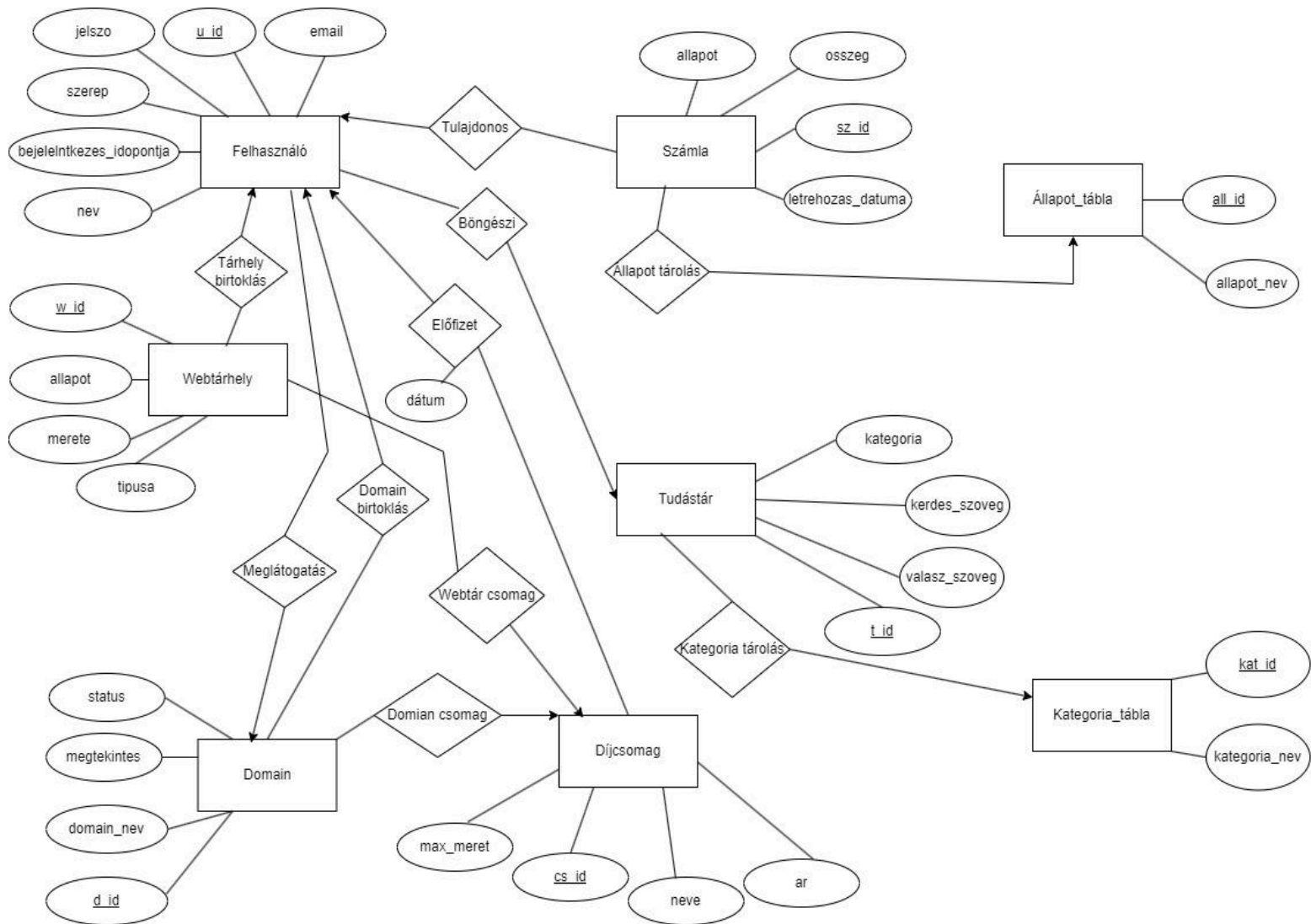
Logikai adatfolyam 2.:



Egyedmodell:



Egyed-kapcsolat:



Séma: (PK = Elsődleges kulcs, FK = külső kulcs)

Felhasználó (u_id [PK], jelszo, email, nev, szerep, bejelentkezes_idopontja)

Webtárhely (w_id [PK], allapot [FK], meret, tipus, u_id [FK], cs_id [FK])

Domain (d_id [PK], domain_nev, status, megtekintes, u_id [FK], cs_id [FK])

Díjcsomag (cs_id [PK], neve, max_meret, ar)

Számla (sz_id [PK], letrehozas_idopontja, osszeg, all_id [FK], u_id [FK])

Tudástár (t_id [PK], categoria [FK], keres_szoveg, valasz_szoveg)

Előfizet (u_id [PK] [FK], d_id [PK] [FK], datum)

Állapot (all_id [PK], allapot_nev)

Kategória (kat_id [PK], categoria_nev)

3NF-re hozás:

1NF-ben van, mert : Minden attribútum atomi.

2NF-ben van, mert: Minden nem kulcs attribútum teljes függőségben van az elsődleges kulccsal.

3NF-ben van mert: Nincs tranzitív függés.

Egyed-esemény-mátrix

[illegible]

Szerep-funkció mátrix

[illegible]

Funckiónális megvalósítás

Használt technológiák:

- Frontend: Svelte (JavaScript Web FrameWork)
- Backend: FastAPI (Python alapú Web FrameWork)
- Adatbázis: Oracle Cloud
- Verziókövetés: Github
- Diagramm készítés: Draw.io
- IDE: Visual Studio, PyCharm
- Dokumentum készítés: Google Docs

Megvalósított funkciók:

Felhasználókezelés: Regisztráció, bejelentkezés, felhasználói profilok

kezelése, jogosultságok beállítása: A rendszer képes felhasználót regisztrálni, bejelentkeztetni. A folyamat jelszó titkosítás valamint autentikációs tokenek kísérik ezzel biztosítva a jogosultság kezelést is az elérhető API pontokban. Eljárással frssítésre kerül. [\[P\]](#)

Domainek kezelése : A rendszer képes domain egyed tárolni és ezt képesek vagyunk CRUD valamint egyéb műveletekkel manipulálni. Triggerel kezeljük az egyediséget. A megtekintés attributumot egy eljárással kezeljük. [\[T\]](#)

Webtárhely kezelése: A rendszer webtárhely egyedet tárol és ezt képesek vagyunk CRUD műveletekkel manipulálni.

Díjcsomagok (különböző tárhely- és adatbázis-kapacitás): A rendszer díjcsomag egyedet tárol ez határozza meg a számlázást és a webtárhely, domainek változását.

Számlakészítés : Képesek vagyunk számlát létrehozni egy előfizetés megtörténetekor. [\[T\]](#)

Tudástár (gyakran ismételt kérdések és válaszai a webhostinggal

kapcsolatban): A felhasználói felületen elérhető a tudástár amiben a kérdések és válaszok találhatók.

Számlák mentése havonta, számlák állapotváltozásainak kezelése az

adatbázisba: A számlákat havonta minden hónap 1.-én automatikusan elkészíti a rendszer. Kezeli a fizetési határidő alapján az állapotokat valamint a számlát ki is lehet egyenlíteni. Triggerel kezeljük, hogy törölt állapotba is állítsuk. Egy eljárással kezeljük a havi kiállítást. [\[T\]](#), [\[P\]](#)

Bevételi statisztika, kimutatás készítés: A felhasználói felületen látható egy statisztika, itt megtalálható a bevétel is havi bontásban. [\[Nézettábla\]](#)

Korlátozás, esetleges kitiltás fizetési késedelem estén : A rendszer kezeli, hogyha a felhasználó nem időben fizet. Egy bizonyos idő után letiltja a vásárolt domainjeit és webtárhelyét. Triggerel és eljárásokkal kezeljük a késés állapot változást valamint a kitiltást. [\[T\]](#), [\[P\]](#)

Legnézettebb oldalak kiszűrése: A felhasználói felületen látható egy statisztika, itt megtalálható a megtekintés alapján rendezett domainek [\[Nézettábla\]](#)

.

Legaktívabb felhasználók kiszűrése: A felhasználói felületen látható egy statisztika, itt megtalálható a domainek, webtárhelyek, megtekintések és számlák alapján rendezett felhasználók. [\[Nézettábla\]](#)

Legtöbbet fizető tulajdonosok kiszűrése: A felhasználói felületen látható egy statisztika, itt megtalálható az összesített fizetett számlák összege alapján rendezett felhasználók. [\[Nézettábla\]](#)

Megvalósított triggerek:

1. Díjcsomag-ajánlás (*TRIGGER*)

- **Funkció:** Minden bejelentkezésnél (felhasználó `bejelentkezés_idopontja` frissül), a rendszer kiszámolja a megtekintések és tárhelyhasználat alapján az optimális díjcsomagot.
- **Trigger neve:** `ajanlas_felhasznalo_trigger`
- **Megjegyzés:** A `felhasznalo` táblán történő frissítésre aktiválódik.

2. Domain név egyediség (*TRIGGER*)

- **Funkció:** Minden domain létrehozáskor a trigger ellenőrzi, hogy a `domain_nev` mező már szerepel-e az adatbázisban.
- **Trigger neve:** `domain_nev_ellenorzes`
- **Megjegyzés:** Elutasítja a beszúrást hibajelzéssel, ha nem egyedi.

3. Késedelmes számla állapotváltás (*TRIGGER*)

- **Funkció:** 2 hónap után automatikusan "késedelmes" állapotra állítja a `szamla` rekordot, ha nem lett kifizetve.
- **Trigger neve:** `szamla_fizetes_keses`
- **Használat:** `AFTER UPDATE` vagy eljárás meghívásán keresztül aktiválható.

4. Számla létrehozása előfizetés esetén (*TRIGGER*)

- **Funkció:** Automatikusan számlát generál, ha új előfizetés (domain vagy webtárhely) történik.
- **Trigger neve:** `szamla_letrehozas`
- **Használat:** `AFTER UPDATE` az előfizet táblán automatikusan létrehoz egy számlát is erről.

5. Számla töröltre állítása (*TRIGGER*)

- **Funkció:** Ha 12 hónapja kifizetett számla, akkor az állapotot "töröltre" állítja. A trigger vagy időzített eljárással aktiválható.
- **Trigger neve:** `szamla_torolt_allapotra`
- **Használat:** `AFTER UPDATE` vagy eljárás meghívásán keresztül aktiválható.

Megvalósított eljárások:

1. Bejelentkezés időpont frissítése (*PROCEDURE*)

- **Funkció:** A login folyamat után frissíti a `bejelentkezes_idopontja` mezőt.
- **Eljárás neve:** `bejelentkezési_idopont_frissítése`
- **Meghívás helye:** `/auth/login` API

2. Fizetési hátralék kezelése, blokkolás (*PROCEDURE*)

- **Funkció:** Ha legalább 6 hónapja nem fizetett, akkor a `domain` és `webtarhely` rekordok blokkolásra kerülnek.
- **Eljárás neve:** `fizetesi_hatralek_kezelese`
- **Automatikus végrehajtás:** Időzítve hívja a rendszer.

3. Számla kiállítás minden hónapban (*PROCEDURE*)

- **Funkció:** Minden hónap 1-jén új számlát állít ki az aktív előfizetésekre.
- **Eljárás neve:** `haviszamlak_kiallitas`
- **Automatikus végrehajtás:** Időzítve hívja a rendszer.

4. Megtekintések növelése (*PROCEDURE*)

- **Funkció:** A domain megtekintés száma automatikusan nő, amikor a felhasználó lekér egy oldalt.
- **Eljárás neve:** `noveli_megtekinteseket`
- **Automatikus végrehajtás:** Időzítve hívja a rendszer.

5. Számla állapotok kezelése (*PROCEDURE*)

- **Funkció:** Ellenőrzi a lejárt számlákat, és frissíti az állapotukat "késedelmes" vagy "törölt" állapotra.
- **Eljárás neve:** `szamlak_ellenorzes`
- **Megjegyzés:** A `szamla_torolt_allapotra,szamla_fizetes_keses` trigger ezt hívja segédként.

Megvalósított lekérdezések:

A lekérdezéseket nézettáblákban tároltuk, ezeket funkcionálnak statsztikaként a felhasználói felületen.

1. NÉZETTÁBLA NEVE: LEGTOBBET_FIZETOK

- **Leírás:** Az összesíti és listázza a legtöbbet fizetett felhasználókat
- **SQL:**

```
CREATE OR REPLACE VIEW legtobbet_fizetok AS
SELECT
  f.nev AS felhasznalo_nev,
  SUM(sz.osszeg) AS osszes_fizetes
FROM felhasznalo f
JOIN szamla sz ON f.u_id = sz.u_id
WHERE sz.all_id IN (2, 4) -- csak lezárt / teljesített
GROUP BY f.nev
ORDER BY osszes_fizetes DESC;
```

2. NÉZETTÁBLA NEVE: LEGNEZETTEBB_DOMAIN

- **Leírás:** Akik a legtöbb domaint és webtárhelyet vettek.
- **SQL:**

```
CREATE OR REPLACE VIEW legnezettebb_domain AS
SELECT
  f.nev AS felhasznalo_nev,
  SUM(sz.osszeg) AS osszes_fizetes
FROM felhasznalo f
JOIN szamla sz ON f.u_id = sz.u_id
WHERE sz.all_id IN (2, 4) -- csak lezárt / teljesített
GROUP BY f.nev
ORDER BY osszes_fizetes DESC;
```

3. NÉZETTÁBLA NEVE: **HAVI_BEVETEL_STAT**

- **Leírás:** Az összesített bevétel havonta
- **SQL:**

```
CREATE OR REPLACE VIEW havi_bevetel_stat AS SELECT
    TO_CHAR(letrehozás_datuma, 'YYYY-MM') AS honap,
    SUM(összeg) AS bevetel
FROM számla
JOIN állapot_tábla állapot ON sz.all_id = állapot.all_id
WHERE all_id IN (2, 4) -- csak fizetett/lezárt számlák
GROUP BY TO_CHAR(letrehozás_datuma, 'YYYY-MM')
ORDER BY honap;
```

4. NÉZETTÁBLA NEVE: **LEGAKTIVABB_FELHASZNALOK**

- **Leírás:** A felhasználók listázása az általuk birtokoltak alapján.
- **SQL:**

```
CREATE OR REPLACE VIEW legaktivabb_felhasznalok AS
SELECT
    f.nev AS felhasznalo,
    COUNT(DISTINCT d.domain_nev) AS domainek_szama,
    COUNT(DISTINCT w.w_id) AS webtarhelyek_szama,
    COUNT(DISTINCT sz.sz_id) AS szamlak_szama
FROM felhasznalo f
LEFT JOIN domain d ON f.u_id = d.u_id
LEFT JOIN webtarhely w ON f.u_id = w.u_id
LEFT JOIN számla sz ON f.u_id = sz.u_id AND sz.all_id IN (2, 4)
GROUP BY f.nev
ORDER BY domainek_szama DESC;
```