**KiwiClub webalkalmazás**

**Fejlesztők:**

Solymosi Richárd

Szabó Tamás

Lauber Attila

Tartalom

[A webalkalmazás specifikációja 3](#_Toc118656901)

[Domain osztályok: 3](#_Toc118656902)

[Service osztályok: 6](#_Toc118656903)

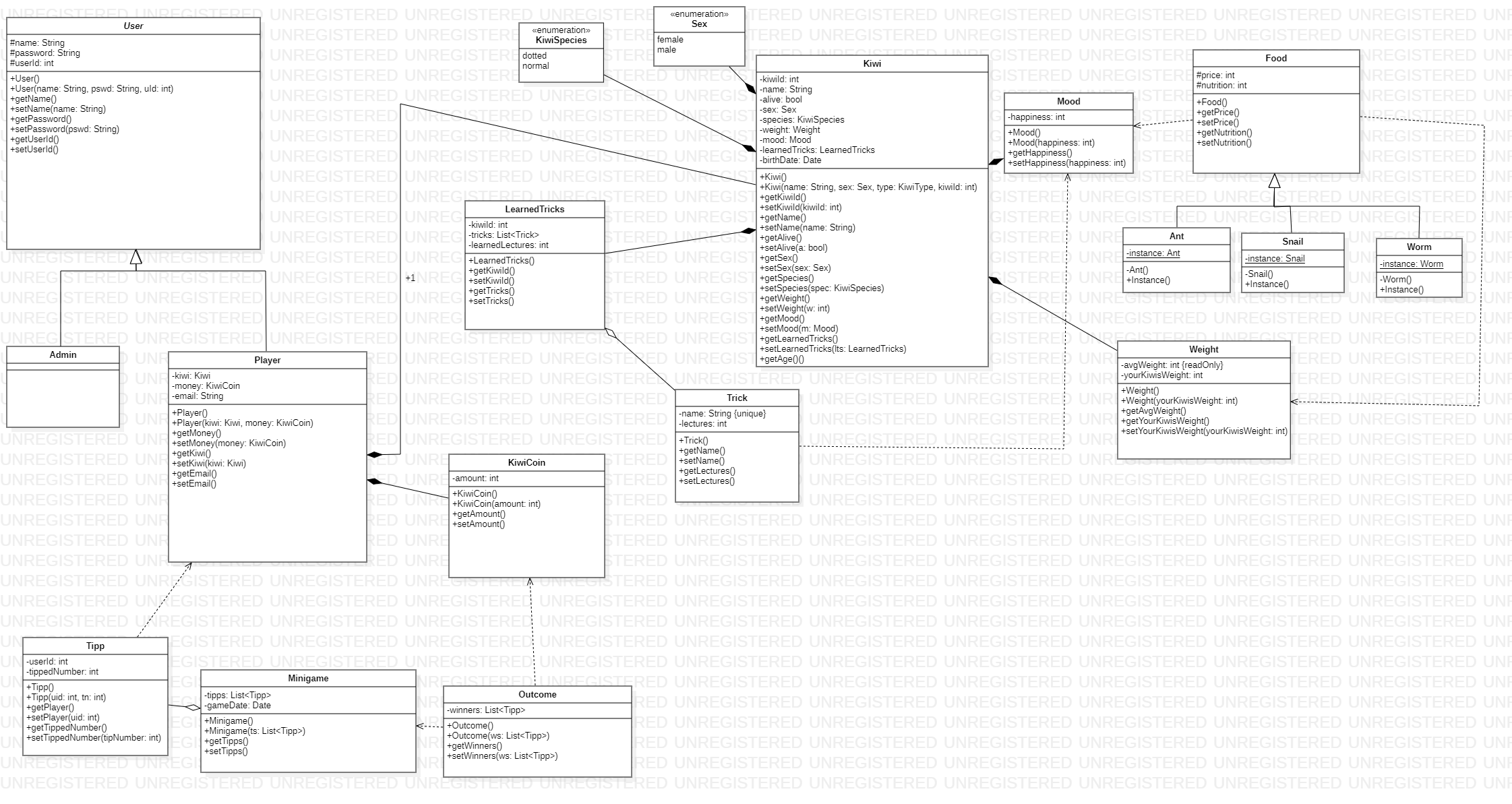
[Perzisztencia réteg: 7](#_Toc118656904)

[View réteg: 8](#_Toc118656905)

A webalkalmazás specifikációja

# A webalkalmazás specifikációja

## Domain osztályok:



* User osztály
  + absztrakt osztály, amelyből az Admin, és a Player osztályok származnak
  + mezők:
    - #name: a felhasználó neve
    - #password: a felhasználó jelszava
    - #userId: a felhasználó egyedi azonosítója
  + metódusok
    - +isAdmin(): absztrakt metódus, amelyet az alosztályok felülírnak, alapértéke false
    - minden mezőhöz setter,getter
  + konstruktorok
    - +User()
    - +User(name: String, password: String, uId: int)
* Admin osztály
  + Az Admin service-t használja, szerkeszteni tudja az oldalt
  + metódusok:
    - +isAdmin() : visszatérési értéke true
  + konstruktorok:
    - +Admin()
* Player osztály
  + A Player service-t használja, csak használni tudja az oldalt
  + mezők
    - -kiwi: A játékos kisállata, amelyet gondozhat
    - -friendsList: A játékos barátait tartalmazó lista
    - -money: A játékos pénze
  + metódusok
    - minden mezőhöz setter,getter
  + konstruktorok
    - +Player()
    - +Player(kiwi: Kiwi, money: KiwiCoin)
* Kiwi osztály
  + A játékos kiwije
  + mezők
    - -trickList: A kiwi megtanult trükkjeit tartalmazó lista
    - -name: A kiwi neve
    - -sex: A kiwi neme
    - -species: A kiwi fajtája
    - -weight: A kiwi súlya, amely a tápláltságtól függ
    - -mood: A kiwi hangulata, amely a tápláltságtól, és a napi tanulástól függ
    - -alive: A kiwi életben van-e
    - -kiwiId: A kiwi egyedi azonosítója
    - -lastFeedDate: Date a kivi legutolsó etetésének a napja
  + metódusok
    - minden mezőhöz setter, getter
  + konstruktorok
    - +Kiwi()
    - +Kiwi(name: String, sex: Sex, type: KiwiType, kiwiId: int)
* LearnedTricks osztály
  + egy adott id-jú kiwi megtanult trükkjeit leíró osztály
  + mezők
    - -kiwiId: A kiwi egyedi azonosítója
    - -tricks: A kiwiId id-jú kisállat megtanult trükkjeit tartalmazza
  + metódusok
    - minden mezőhöz setter, getter
  + konstruktorok
    - +LearnedTricks()
* Trick osztály
  + Trükk, amit a kiwi megtanulhat
  + mezők
    - -name: A trükk neve
    - -lectures: Ennyi leckéből áll a trükk
  + metódusok
    - minden mezőhöz setter, getter
  + konstruktorok
    - +Trick()
* Sex enum
  + A kiwi neme, a felületen vizuálisan megjelenik
* KiwiType enum
  + A kiwi fajtája, a felületen vizuálisan megjelenik
* Mood osztály
  + A kiwi kedvét mutató osztály, amely befolyásolja a napi tanulást, és azt az etetés
  + mezők
    - -happines: A kiwi kedvét reprezentálja (1-100 közötti érték)
  + metódusok
    - A happines mezőhöz getter
  + konstruktorok
    - +Mood()
* Food osztály
  + absztrakt osztály amelyből a különféle táplálék osztályok származnak
  + mezők
    - #price: Az étel ára
    - #nutrition: az étel tárpértéke
  + metódusok
    - get-set metódusok a price és a nutrition mezőkhöz
  + konstruktorok
    - +Food()
* Ant osztály
  + A Food-ból származó singleton osztály
  + mezők
    - -instance az osztály egyetlen példányát tartalmazza, mivel az osztály singleton
  + metódusok
    - +Instance() az osztály egyszeri példányosításáról gondoskodik
  + konstruktorok
    - -Ant()
* Snail osztály
  + A Food-ból származó singleton osztály
  + mezők
    - -instance az osztály egyetlen példányát tartalmazza, mivel az osztály singleton
  + metódusok
    - +Instance() az osztály egyszeri példányosításáról gondoskodik
  + konstruktorok
    - -Snail()
* Worm osztály
  + A Food-ból származó singleton osztály
  + mezők
    - -instance az osztály egyetlen példányát tartalmazza, mivel az osztály singleton
  + metódusok
    - +Instance() az osztály egyszeri példányosításáról gondoskodik
  + konstruktorok
    - -Snail()
* Weight osztály
  + A Kiwi osztályban jelenik meg mint adattag, minden kiwinek van különsúlya, amit százalékos formában lesz értelmezve, ezért az átlag súly adattag
  + mezők
    - -avgWeight: egy átlag kiwi testsúlyát írja le (beégetett érték)
    - -yourKiwisWeight: a játékos kiwijének aktuális súlya
  + metódusok
    - get metódusok az adattagokhoz
    - setYourKiwisWeight(yourKiwisWeight: int)
* KiwiCoin osztály
  + Játékbeli pénz, amelynek mennyiségét befolyásolja az ételvásárlás, a minijáték
  + mezők
    - -amount: A pénz mennyisége
  + metódusok
    - minden mezőhöz getter,setter
  + konstruktorok
    - +KiwiCoin()
    - +KiwiCoin(amount: int)
* MiniGame osztály
  + A számkitalálós játékot reprezentálja
  + mezők
    - -tipps: A különböző játékosoktól beérkezett tippek listája
  + metódusok
    - get-set metódusok
  + konstruktorok
    - Minigame()
    - Minigame(ts:List<Tipp>)
* Outcome osztály
  + A minijáték nyerteseit tartalmazza, valamint befolyásolja a nyertes felhasználók pénzét
  + mezők
    - -winners: A minijáték nyertesei
  + metódusok
    - get-set metódusok
  + konstruktorok
    - +Outcome()
    - +Outcome(ws:List<Tipp>)
* Tipp osztály
  + Egy játékostól beérkező tipp, amelyet a MiniGame osztály tárol el
  + mezők
    - -userId: A játékos azonosítója akitől a tipp érkezett
    - -tippedNumber: A tippelt szám
  + metódusok
    - get-set metódusok
  + konstruktorok
    - +Tipp()
    - +Tipp(uid:int,tn: int)

## Service osztályok:

A képen szöveg látható

Automatikusan generált leírásA képen szöveg látható

Automatikusan generált leírásA képen szöveg látható

Automatikusan generált leírásA képen szöveg látható

Automatikusan generált leírásA képen szöveg látható

Automatikusan generált leírás

Perzisztencia réteg:A képen szöveg látható

Automatikusan generált leírás

* CreditentialRepository
  + játékosok belépési adatait tároljuk (és az egy adminét):
    - userId: int
    - name: String
    - password: String
  + metódusok:
    - getPlayerByName()
* PlayerRepositry
  + a játékosok játékhoz kapcsolódó adatait tároljuk el
    - kiwi: Kiwi (one to one kapcsolat)
    - money: KiwiCoin
    - email: String
  + metódusok:
    - getPlayerByName()
    - getPlayerById()
* KiwiRepository
  + kivik minden adatát tároljuk:
    - kiwiId: int
    - name: String
    - alive: bool
    - sex: Sex
    - species: KiwiSpecies
    - weight: Weight
    - mood: Mood
    - birthDate: Date
    - learnedTricks: LearnedTricks
  + metódusok:
    - getKiwiById()
* TricksRepository
  + ez a repository tárolja a játékban elérhető trükköket, amiket a kivi megtanulhat
  + tároljuk
    - -name: String; a trükk neve, ez egyedi, primary key
    - -luctures: int a trükk megtanulandó leckéinek a száma
  + metódusok
    - +getTricksByName()
* LearnedTricksRepository
  + ez tárolja kivik által megtanult trükköket
    - kiwiId: int
    - trickName: String
    - learnedLecutres: int
  + metódusok:
    - getLearnedTricksByKiwiId()
* MinigameRepository
  + a játékban lesz egy mini játék, amivel a játékos pénzt tud szerezni, a játék lényege, hogy tippelni kell egy számot és ha szerencséje van az illetőnek, akkor nyerhet pénzt és ezeket a tippeket kell eltárolnia a reponak
    - tipps: Tipp-ek gyűjteménye
    - gameDate: Date -> ez alapján lesz szűrve, hogy melyik napi játék
  + metódusok:
    - getTippsByGameDate()

## View réteg:

* **az oldalak amiket a felhasználó látni fog:**
  + **login**
  + **regisztráció**
  + **kivi adatlap/kezdőlap**
    - **etetés**
    - **tanítás**
  + **felhasználói adatok módosítása**
  + **minigame**