Eötvös Loránd Tudományegyetem

Informatikai Kar

Catan társasjáték

Szakdolgozat

*Készítette: Konzulens:*

Fodor Kristóf XXX

RFXQHM, titulus,

Programtervező informatikus BSc –B szakirány tanszék

Budapest

2023

Tartalomjegyzék

[1. Bevezetés 3](#_Toc135680112)

[2. Felhasználói dokumentáció 4](#_Toc135680113)

[2.1. A játék futtatásához szükséges gépigény, programok 4](#_Toc135680114)

[2.2. A játék kezelőfelülete 4](#_Toc135680115)

[3. Fejlesztői dokumentáció 5](#_Toc135680116)

[3.1. Felhasznált technológiák a fejlesztéshez 5](#_Toc135680117)

[3.1.1 IntelliJ IDEA 5](#_Toc135680118)

[3.1.2 Kotlin 5](#_Toc135680119)

[3.1.3 Ktor 5](#_Toc135680120)

[3.1.4 Kotest 5](#_Toc135680121)

[3.1.5 Gson 5](#_Toc135680122)

[3.1.6 Hibernate 5](#_Toc135680123)

[3.1.7 TornadoFX 6](#_Toc135680124)

[3.1.8 Gradle 6](#_Toc135680125)

[3.2. Áttekintés, programok felépítése rövid leírása 6](#_Toc135680126)

[3.2.1 Kliens 6](#_Toc135680127)

[3.2.2 Szerver 6](#_Toc135680128)

[3.2.3 SDK (Software Development Kit) 6](#_Toc135680129)

[3.3. SDK 6](#_Toc135680130)

[3.3.1 Főbb osztályok áttekintése 6](#_Toc135680131)

[3.3.2 Osztályok diagramjia 10](#_Toc135680132)

[3.4. Kliens 10](#_Toc135680133)

[3.4.1 Kilens -> Szerver kommunikáció 11](#_Toc135680134)

[4. Befejezés 14](#_Toc135680135)

# Bevezetés

A mai világban nagyon sokan szeretnek társasjátékozni a gyerekek és a felnőttek körében is. A 2019-ben kitörő COVID-járvány miatt a szociális kapcsolatok megszűntek, mindenki távolságot tartott, bezárkózott otthonába. Ezért sajnos a baráti társaságok, nem egy háztartásban élő rokonok közötti társasjátékos alkalmak száma a pandémia idején gyakorlatilag nullára csökkent. Ebben az időszakban terjedtek el a különböző online felületek, ahol minden biztonságosan a saját otthonából játszhatott különböző társasjátékokkal. Ha a játék során valamilyen platformon videóhívásban vannak a résztvevők, akkor az elég jól meg tudja közelíteni a személyes találkozókat.

A szakdolgozatom témájának a Catan társasjátékot választottam. Ez a játék volt az, amely rabul ejtett, bevezetett a modern társasjátékok világába. A Catan társasjátékot Klaus Teuber tervezte, és 1995-ben adta ki először Németországban a Kosmos cég[[1]](#footnote-1). A világ számos nyelvére lefordították, ezáltal nagyon sok ember által kedvelt játékká vált. Ez bizonyítja az is, hogy több díjat is nyert a játék. Ezek közül a legkiemelkedőbbet rögtön a kiadás évében nyerte el, a társasjáték világ egyik legrangosabb díját, a Spiel des Jahres díjat[[2]](#footnote-2).

A Catan egy olyan stratégiai játék, amely során egy hatszöglapkákból álló lakatlan szigetet próbálnak a játékosok belakni településekkel, városokkal, utakkal oly módon, hogy lehetőleg minél kedvezőbben férjenek hozzá az építkezéshez szükséges nyersanyagokhoz, ezáltal minél több győzelmi pontot szerezzenek. A játékosok a saját körükben kereskedhetnek – ez történhet a bankkal, de a játékostársaikat is megpróbálhatják alkura bírni.

Ide szépen le kell írni tömören milyen formában valósítottad meg a programot.

A következőkben először a fejlesztői majd a felhasználói dokumentációt szeretném bemutatni.

# Felhasználói dokumentáció

## A játék futtatásához szükséges gépigény, programok

## A játék kezelőfelülete

# Fejlesztői dokumentáció

Ez a fejezet a szakdolgozat programjait fogja részletezni fejlesztői szemszögből, hogy megértse a működését.

## Felhasznált technológiák a fejlesztéshez

### IntelliJ IDEA

Az IntelliJ IDEA a JetBrains készítette Java és Kotlin programozók számára készített IDE, ami specifikusan JVM alapú nyelvekhez lett fejlesztve. Képes használni Gradle-t és Maven-t is, van egy beépített debugger.

### Kotlin

A Kotlin egy JVM alapú, statikusan erősen típusos, magas szintű, JetBrains által fejlesztett programozási nyelv. Egyik nagy előnye, hogy egyszerre lehet használni Kotlin-t és Java-t is ugyan abban a kódbázisban, ezért lehetséges az, hogy egy meglévő kódbázis fokozatosan áll át Java-ról Kotlin-ra.

### Ktor

A Ktor az egy aszinkron alapú kommunikációs könyvtár, ami képes http készíteni/kiszolgálni, vagy akár Socket/WebSocketeken keresztül kommunikálni. Ktor-t az elejétől kezdve úgy fejlesztették, hogy Kotlinnal legyen használva, egyszerűen lehet hozzá készíteni egyedi plugin-okat. Ktor segítségével van megoldva a kliens és szerver közti socket kommunikáció.

### Kotest

A Kotest egy teszt keretrendszer Kotlinban írt programok tesztelésére készült. Több féle tesztelési stílust támogat, GIVEN WHEN THEN stílusú teszteket lehet vele készíteni, vagy csak sima egyszerű funckiót futtat le.

### Gson

A Gson egy nyílt forráskódú Java könyvtár, amellyel Java Object-eket lehet Json-nel reprezentálni, és Json által reprezentált objektumot lehet Java Object-é alakítani. Ezt a könyvtárt használtam arra, hogy a kliens és a szerver tudjon egymással kommunikálni.

### Hibernate

A Hibernate egy ORM (Object Relation Mapping) eszköz, ami képes arra, hogy Java object-et egy adatbázisban lévő elemhez rendel. Képes adatbázisba menteni, kiolvasni és változtatni.

### TornadoFX

A TornadoFX egy kotlin wrapper a Java FX-hez, ami megkönnyíti kotlinban fejlesztett ablakos alkalmazások fejlesztését.

### Gradle

TODO

## Áttekintés, programok felépítése rövid leírása

A projekt több különböző alkalmazásból áll mindegyiknek megvan a saját feladata

### Kliens

Egy grafikus felület, ami megjeleníti a szervertől kapott információkat a képernyőt, a felhasználó ezzel kommunikál a szerverrel.

### Szerver

Kiszolgálja a klienst vagy klienseket a tőlük kapott információkkal. A szerverhez egyszerre több kliens is csatlakozhat.

### SDK (Software Development Kit)

A kliens és szerver közös komponenseit tárolja, megkönnyíti a fejlesztés olyan szempontból, hogy közös komponenseket egy helyen kell csak megváltoztatni.

## SDK

Az SDK-nak a feladata, hogy a közös komponenseket tárolja, Kotlin-ban lehetőség van arra, hogy olyan osztályokat használjunk fel másik projektekből. Így nincs szükség ugyanazt az osztályt több különböző helyen deklarálni, ezzel csökkentve a kódbázis méretét, és így nem keletkezhet olyan hiba, hogy a kliens és a szerver máshogy implementálja ugyanazokat az osztályokat. Ha egy osztályt módosítani, vagy létre kell hozni akkor azt csak egy projektbe kell.

### Főbb osztályok áttekintése

#### Map osztály

A catan társas játékban a pálya hatszögekből áll, ezért nem lehet egyszerűen ábrázolni, a Map osztály felel azért, hogy a generált pályán a mezők (Tile-ok) helyes helyen legyenek egymáshoz képest, és ezeket a Tile-okat össze kapcsolja, a Tile-oknak a 6 sarkára rakjon egy Vertex osztályt, és ezeket a Vertexeket összekösse Edge osztállyal. Opció, hogy egy előre elkészített pályát generáljon le, ahol a Tile-okhoz hozza van rendelve a dobott szám és a típusa.

Alternatívan lehet Map osztály elő állítani egy MapDto-ból ami tartalmazza tömören a Tile-okat és ugyan azt a pályát állítsa elő, mint ha manuálisan generáltuk le.

##### Map generálása

A pálya generálása két lépésből áll

* Mezők generálása
* Mezők és csúcsok össze kötése

Mezők generálásához le kell generálni alap esetben 19 Tile-t. A Tile-nak az id-ja két részből áll, az elején egy prefix ’’T’’ és a prefix után a suffix-ja, pedig ahányadik generált Tile. Például az első Tile ’’T0’’ második ’’T1’’ .

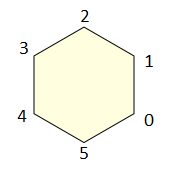
Majd a második lépésben ezeket a Tile-okat kell összekötni. Ezt úgy teszem meg, hogy végig iterálok tiles tömbön, és a jelenlegi tilehoz hozzá kötöm a tőle jobbra, jobbra lefele és balra lefele lévő Tile-okat. Ezeket a Tile-okat úgy kapom meg, hogy a jelenlegi Tile id-ját átalakítom Coordinate-té és „getCoordBottomRight / getCoordBottomLeft / getCoordRight” metódusok segítségével megkapom a relatív koordinátát. A kapott koordinátát átváltam toTile() segítségével Tile-lá, és az attachTileToSide segítségével össze tudom őket kötni.

##### Függvények

|  |  |
| --- | --- |
| generateTiles(numberOfRows: Int, isGeneric: Boolean) | Legenerálja az összes Tile-t, ha az isGeneric true, akkor beállít egy előre készített pályát. Pálya legelső és utolsó sora 3 nagyságú |
| attachAllTiles() | Az összes Tile-t ami ebben a Map-ben található össze kapcsolja az oldalai mentén. |
| addEdges | A tile-okon található vertexeket összekötjük egymással |
| toDto(): MapDto | Összeállít egy olyan MapDto-t amiből össze lehet állítani ugyanezt a Map objektumot |
| attachTileToSide(  attachToThis: Tile,  bottomRight: Tile,  right: Tile,  topRight: Tile,  topLeft: Tile,  left: Tile,  bottomLeft: Tile,  ) | Az attachToThis-hez oda köti a Tile-okat, olyan irányból amilyen irányból megadjuk. |
| idToCordinate(id: Int) | A tile ID-jából össze állít egy kordinátát ami egy pár számból áll, a pár első része a sorszámra vonatkozik, a második része a sorban lévő helyére. |
| getCoordBottomRight(cord: Coordinate): Coordinate | Vissza adja a ’’cord’’-hoz képest a jobbra lent lévő kordinátát |
| getCoordBottomLeft(cord: Coordinate) : Coordinate | Vissza adja a ’’cord’’-hoz képest a balra lent lévő kordinátát |
| getCoordRight(cord: Coordinate) : Coordinate | Vissza adja a ’’cord’’-hoz képest a jobbra lévő kordinátát |

#### Tile osztály

Ez az osztály reprezentálja a Catan mezőit. Catan egy mezője hatszögből áll. Az osztályban van egy vertices lista ami 6 Vertex elemet tartalmaz, az 1. ábrán látható, hogy a lista elemei melyik csúcsot reprezentálják.



. ábra

##### Függvények

|  |  |
| --- | --- |
| connectVertices(edges: List<Edge>) | Összeköti a Vertex-eket a fent ábra alapján majd a keletkezett éleket a paraméterként megadott tömbbe helyezi |

#### Player osztály

Ez az osztály felel egy aktív játékban lévő játékos reprezentálására.

##### Függvények

|  |  |
| --- | --- |
| Függvény név és leírása | Működése |
| calculateMyPoints(): Int | Kiszámolja a játékos pontjait, beleszámítva a ki nem játszott pont kártyákat. |
| toDto(hidden: Boolean): PlayerDto | Átalakítja a Player osztályt egy PlayerDto osztállyá, amit a szerver tovább tud küldeni. A hidden paraméter, ha igaz akkor olyan dto-t ad vissza amin a játékos kártyái és nyersanyagai elvannak tüntetve |
| refreshFromDto(dto: PlayerDto) | A paraméterül kapott dto alapján felfrissíti a játékos adatait. |

### Osztályok diagramiai

A képen szöveg, képernyőkép, Betűtípus, dokumentum látható

Automatikusan generált leírás

## Kliens

A Kliens egy grafikus felület ami a szervertől kapott információkat jeleníti meg olyan formában, hogy a felhasználó láthassa és ezen keresztül kommunikáljon a szerverrel.

### Kilens -> Szerver kommunikáció

#### Játékon kívüli kommunikáció

A képen szöveg, képernyőkép, kör, Betűtípus látható

Automatikusan generált leírás

* Regiszrtáció: A felhasználó készíthet egy új felhasználót, amivel betud majd jelentkezni.
* Bejelentkezés: A felhasználó bejelentkezhet egy már regisztrált felhasználóval.
* Játék szoba létrehozása: Létre hozhatsz egy várakozó szobát, amibe betud több játék csatlakozni.
* Elérhető várakozó szobák megtekintése: Letudod a szervertől kérni a jelenleg elérhető várakózó szobákat
* Belépés várakozó szobába: Egy elérhető várakozó szobába betudsz lépni
* Játék elindítása: A várakozóban, a szoba tulajdonosa elindíthatja a játékot, és a szobában lévő játékosok átkerülnek a játékba.

#### Játékon belüli kommunikáció

A játékon belüli kommunikációnak két része van, mielőtt minden játékos lerakta a maga faluit és útjait, és miután mindenki lerakta.

##### Mielőtt mindenki mindent lerakott

A képen szöveg, képernyőkép, Betűtípus, fekete látható

Automatikusan generált leírás

Játék elején a kliensnek ki kell választania egy csúcsot, ahova letesz egy falut, és egy élt, ami ehhez a csúcshoz kapcsolódik. Majd ezt a csúcs és él kombinációt küldi tovább a szervernek. Ha minden játékos letett 2 falut, és élet azután elindul ténylegesen a játék.

##### Miután mindenki mindent lerakott

A képen képernyőkép, szöveg, diagram, tervezés látható

Automatikusan generált leírás

Miután elindult a játék ténylegesen a játékos 5 különböző dolgot tud csinálni:

* Passzolás: Elküldi a szervernek, hogy végzett a játékos a körével
* Tengeri csere: Játékos kiválaszt egy nyersanyagot, amiből 4 vagy annál többje van a játékosnak, és átcseréi egy attól különböző nyersanyagra.
* Fejlesztés kijátszása:
  + Út építés: játékos kiválaszt két utat, amit megszeretne építeni betartva a szabályokat, és elküldi a szervernek.
  + Monopólium:
  + Találmány:
  + Lovag:
* Rabló áthelyezés:
* Vásárlás:
  + Út:
  + Falu:
  + Város:
  + Fejlesztés:

## Szerver

A szerver feladata, hogy kiszolgálja a klienst a tőle kapott információk alapján. A felhasználó tudjon regisztrálni

# Befejezés

Ide lehet leírni, hogy hogyan készítenéd el azokat a részeket, amik eddig nem történtek meg.

1. <https://www.catan.com/catan> (Utolsó megtekintés: 2023.04.15. 17:00) [↑](#footnote-ref-1)
2. <https://www.spiel-des-jahres.de/en/spiele/die-siedler-von-catan/> (Utolsó megtekintés: 2023.04.15. 16:45) [↑](#footnote-ref-2)