Beágyazott rendszerek szoftvertechnológiája

Házi feladat

**Tervdokumentáció**

Lövöldözős Játék: Szóbalövő

Készítők:

Hencz András (IWQKZW)

Rácz Benedek (AQAEFR)

Bitay Olivér (R6R1I5)

Konzulens: Györke Péter

# Bevezetés

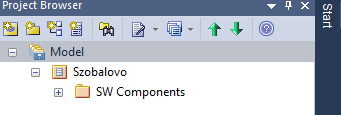
A házi feladatként beadott játékunk célja hiányos szavakba belelőni a megfelelő betűket. A feladatban meg kell valósítani a játék motorját, egy hálózati részt és a GUI-kat.

A játék játszható egy- illetve kétszemélyes módban is, ezért szükség van hálózati kapcsolat kialakítására.

Az UML szerű dokumentációhoz az Enterprise Architect nevű szoftvert használjuk a Sparx Systems-től. Ebben egyszerűen ábrázolhatók az egyes UML komponensek. A program használata nem ingyenes, de letölthető 30 napos próbaverzió. A program hátterében egy ORACLE adatbázis található mely tartalmazza a létrehozott komponenseket és a közöttük lévő kapcsolatokat is! Az Enterprise Architect segítségével modellezhető mind a szoftver statikus mind a dinamikus viselkedése.

# Alapvető struktúrák, osztályszervezés

Mint említettük a Szóbalövő játszható kétszemélyes módban is, ezért szükség van valamiféle hálózati kapcsolatra. A program motorja egy szerveren fut, amelyhez egy kliens segítségével lehet csatlakozni. A kliens feladata létrehozni a menüket és a GUI-t, kapcsolódni a szerverhez majd a játék során a fenntartania a folyamatos kommunikációt.

Első lépésként létrehoztunk a programnak egy modell-t:

A SW Component mappába kerülnek az egyes komponensekhez tartozó class-ok.

A program indulásához szükséges main függvény a MainClass osztály tartalmazza. Ez példányosítja az adott játékhoz tartozó szervert és a klienst.

## Beállítások

Fontos még itt megemlíteni a beállításokat, mert ezeknek is nagy szerepük van az osztályok kialakításában.

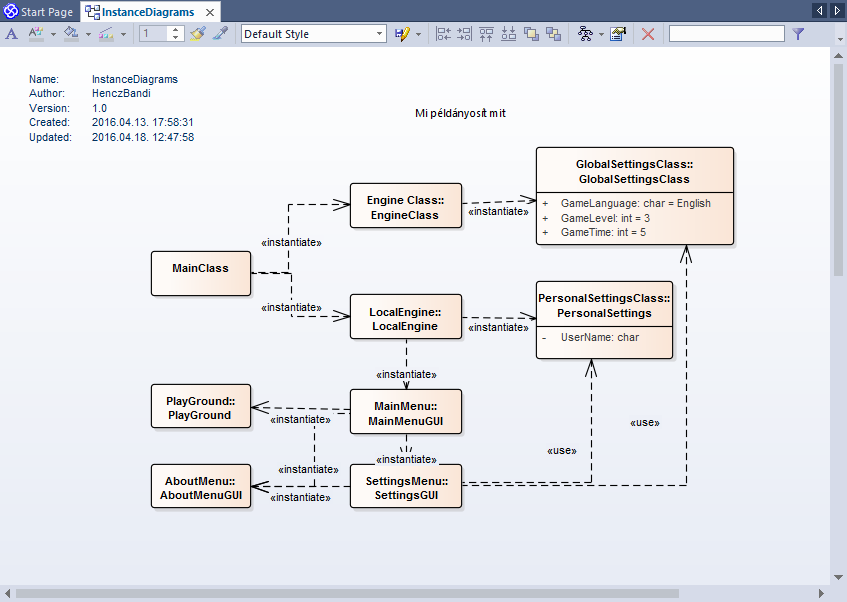
A specifikációban meghatározott beállításokat két különálló részre bonthatjuk szét. Vannak olyanok, amik a játékkal kapcsolatosak, illetve olyan, amely a játékossal. Ez utóbbi a játékos neve, az előbbihez tartozik a játék nehézségi szintje, a szavak nyelve valamint a játékidő.

A játékos nevét természetesen a lokális motornak (tehát a kliensnek) kell ismerni, a többi beállítást pedig a motornak, ami a szerveren fut.

Amikor elindítunk egy új játékot akkor a kliens csatlakozik a megfelelő szerverhez. Egy játékos mód esetén ez a saját maga által elindított szerver a localhoston, multiplayer esetén pedig meg kell határozni, hogy melyik szerverhez csatlakozzon. Természetesen annak a játékosnak a beállításai fognak érvényre jutni kétjátékos mód esetén, amelyiknek a szerveréhez csatlakozik a másik játékos is.

Látható tehát, hogy többféle interfész is szükséges a beállítások elmentésére. Ezeknek létrehoztuk a GameSettings és a GamerSettings osztályokat.

Az alábbi ábrán az látható, hogy melyik osztályt mi példányosít:



Természetesen ez a diagram is megtalálható a csatolt eap fájlban.

# GUI szervezés

Alapvetően a játék eléggé statikus jellegű, a felhasználó beavatkozásai alapján kell feladatokat elvégezni. Ez alapján mondhatjuk, hogy a GUI szerepe a játék vezérlésében kiemelkedő.

Alapvetően kétféle felhasználói felületet kell elkülöníteni. Az egyik a statikus menük, a másik a (játékos beavatkozására) dinamikusan változó játéktér.

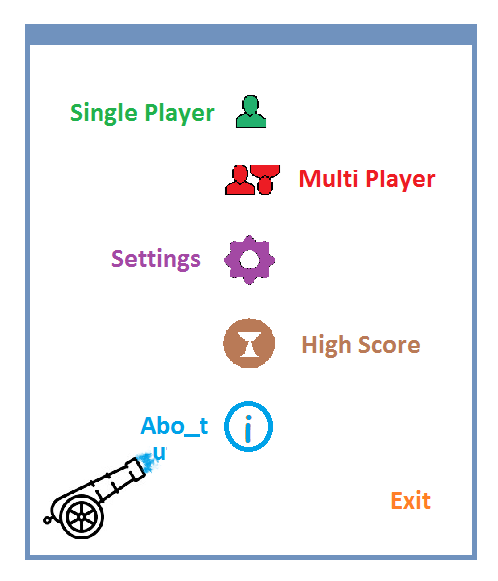
## Menük

Alapvetően háromféle menüt kell megkülönböztetnünk. Az egyik a főmenü (main menu), amely tartalmazza a játékindításhoz szükséges gombokat, a másik a beállítás menü (settings menu) amelyben a beállításokat menthetjük el, illetve a harmadik az about menu.

Ezeket JFrame segítségével tervezzük megvalósítani.

A menüket a kliens példányosítja.

A főmenü terve:



A beállítások menü terve már szerepel ebben a dokumentumban ( XXC.ábra). Az about menü tartalmazza a készítők nevét és elérhetőségét.

## Játéktér

A játéktér a játék egyik legfontosabb és legbonyolultabb eleme, mivel ennek a szerepe a konkrét játék megjelenítése, a játék irányítása, és az eredmények kiírása.

A játéktéren találhatóak a szavak, az ágyú és a belövendő betűk is.

# Szerver, kliens létrehozása

# Hálózati kommunikáció

# Összefoglaló