**ZEUSZ PROJEKT**

Előkészítés

A csapat:

„Kentaur Development Team”

Projektvezető: Dori Bence

Kapcsolattartó: Schönberger Zsófia

Adminisztrátor: Esse Zsolt

Demonstrátor: Balogh Tamás

Megrendelő:

„Szabadítsd fel az erőforrást, ha már nem használod” Társaság Szabadszellemű Magániskolája

**A feladat**

A projekt célja egy olyan szoftverrendszer elkészítése amely az iskola vezetésének, tanárainak, diákjainak nyújt olyan szolgáltatásokat, melyek segítésével elektronikus úton követhetik nyomon és szervezhetik a tanulmányokhoz szükséges feladatokat, valamint nyilvántartást vezessen az intézményben dolgozók és a hallgatók személyi adatairól.

A diákoknak lehetőségük lesz tárgyakat felvenni, leadni vagy elhagyni az iskolát. Módjukban áll lekérdezni teljesített és felvett tárgyaikat, azok érdemjegyeit, nyomon követni az addigi intézményi előéletüket. Megadhatják személyes adataikat, valamint módosíthatják is azokat.

A tanárok vezetői jóváhagyással hozzáadhatnak, illetve törölhetnek saját kurzusaik közül, módosíthatják annak létszámát. Jegyeket írhatnak be és közvetlen üzenetet küldhetnek a tárgyat felvett hallgatóknak.

A vezetés egyben az adminisztrátori feladatkört is betölti, teljes hozzáférést kell biztosítani minden adathoz, tárgyhoz, nyilvántartáshoz.

Minden foglalkoztatottnak (vezető, tanár, diák) a rendszer belépéskor generál egy egyedi belépőkódot és jelszót, melyet az aktuális személynek lehetősége van megváltoztatni.

Egy hallgató az adatbázisban csak egyszer szerepelhet, ismételt jelentkezés esetén törlődnek addigi eredményei. Tárgyakat a hallgató csak aktív státuszú félévkor vehet fel a félév előtti meghatározott időszakban. Ha a hallgató passzív félévre ad be kérelmet a regisztrációs időszak alatt, a rendszer automatikusan lelépteti a már esetlegesen felvett tárgyairól.

A szoftvernek statisztikai funkciókat kell biztosítania a vezetésnek pedig a teljes iskola nyomon követésére, értékelésére.

Lekérdezhető statisztikák:

* Tárgyak teljesítési aránya – kiválasztott félévben, kiválasztott tárgyat, tárgyakat milyen arányban végezték el a hallgatók
* Hallgatói státuszok lekérdezése – a kiválasztott időszakban, időszakokban hány aktív és hány passzív hallgató volt a rendszerben
* Kiiratkozási arány – kiválasztott időszakban, időszakokban mennyi volt a kiiratkozások száma
* Passzív-félév arány - kiválasztott időszakban, időszakokban mennyi volt a passzív kérelmet benyújtók száma

**A szoftver**

1. A rendszer felhasználói és funkciói

A tervezett szoftver felhasználóit három csoportba oszthatjuk, Hallgató, Tanár és Vezető. A rendszer mindhárom csoport számára más-más funkciókat és felületeket biztosít.

Az iskola hallgatói egy központi felületen követhetik nyomon és szervezhetik tanulmányaikat az iskolában.



Lehetőségük lesz **Tárgyakat felvenni** az adott időszak kiválasztásával és a számára felvehető tárgyak listájából kiválaszthatja mely tárgyakat szeretné hallgatni.

A **Felvett tárgyak listázásakor** az aktuális félév kiválasztásával megkapja az arra a félévre felvett kurzusainak a listáját.

Az **Órarend** kiválasztásakor a hallgató a kért félévéről egy átlátható, heti nézetre bontott listát kap saját kurzusairól és azok időpontjairól.

A hallgatónak lehetősége lesz a regisztrációs időszakban a felvett **Tárgyak leadására**, melyet hasonlóan a tárgy felvételhez, a félév kiválasztásával egy listából kijelölheti a leadni kívánt tárgyait.

Megtekinthető lesz az elektronikus **Leckekönyv**, melyben a választott időszak tárgyairól részletes leírást kaphatunk, úgymint: tárgy neve, kódja, oktatója és érdemjegye.

A **Személyes adatok felügyelete** funkciónál a hallgatónak kilistázódik az iskola által tárolt személyes adatai, melyet lehetősége van szerkeszteni. A visszaélések elkerülése végett a hallgató nem tudja elmenteni a változtatásokat, csupán elküldheti az új/módosított adatait jóváhagyásra, ha a vezető megerősíti csak ebben az esetben fog ténylegesen módosulni a kívánt mező.

A **Passzív félév kérelmekor** és **Kilépés az iskolából** funkcióknál a hallgató egy indoklás megadása után jelezheti szándékát a vezetés felé. Kilépéskor a hallgató összes adata átkerül egy archív adatbázisba, melyet a rendszer automatikusan elvégez.

A tanárok szervezik és oktatják az iskola által kínált tantárgyakat, melyek a vezetés jóváhagyásával léphetnek csak életbe.



A **Tárgy hozzáadás**, **Tárgy törlése** és **Tárgy módosítása** funkcióknál a tanárok egy listát kapnak a meglévő tárgyakról, mely segítségével megtekinthetik annak részletes információit, módosíthatják azokat, indoklás megadása után töröltethetik, vagy teljesen új tárgy minden adatának megadásával jelezhetik szándékukat a vezetésnek, akik dönthetnek az adott igényről.

A tanári funkciók közé tartozik az **Értékelés**, melynek segítségével a tanár saját tárgyai, majd annak hallgatóinak listázása után a személyek kiválasztásával megadhatja a diákok által elért érdemjegyeket, melyek mentődnek az adatbázisba.

Minden tanárnak lehetősége van **Üzenetküldésre**, vagy egy címlistából egyesével kiválasztva hozzá járó hallgatóknak és a vezetőknek vagy szűrhet a saját tárgyaira járó hallgatók csoportjára.

A vezetés az iskola irányításáért felel, egyben adminisztrátorok, valamint felelősek az iskola személyi nyilvántartásának karbantartásáért.



A vezetés feladatai közé tartozik a **Kérelmek elbírálása**. Egy időrendi listát kap a vezető az összes hallgatói és tanári igénylésről, melyekre egyesével kattintva eldöntheti, hogy jóváhagyja vagy elutasítja azt. A döntés eredményéről az igénylő felet automatikus üzenet értesíti.

Az **Üzenetküldés** funkció hasonló a tanárokéhoz, azzal a különbséggel, hogy a vezető a teljes címlistát láthatja és az összes tantárgy hallgatóira van lehetősége szűrni és szöveges üzenetet küldeni.

A **Nyilvántartási** adatok kezelésében közvetlen hozzáfér az összes, rendszerbe felvett személy adataihoz és módosíthatja azokat. Új tanár vagy hallgató esetén neki van lehetősége hozzáadni az új tagot a rendszerhez.

**Kimutatás készítésekor** a kívánt időszak, időszakok kiválasztásával valamint a látni kívánt mutatók megjelölésével a rendszer diagramot generál a kapott értékekből.

1. A rendszer szerkezete

A rendszer felhasználói három felületet használnak, így az általuk igénybe vett funkciók három fő részre oszthatóak, mint

* Hallgatói funkciók
* Tanári funkciók
* Vezetői funkciók

Ezt a modulrendszert alapul véve a szoftver tervezését, azok elemeiből indulva végeztük.



A hallgatók számára nyújtott lehetőségek megvalósításához szükséges a grafikus **Hallgatói kezelőfelület**, ezen keresztül kommunikálhatnak a rendszerrel. Három fő részre bonthatóak a tevékenységeik, melyeket **a Személyi adat kezelő**, a **Tantárgy kezelő** és a **Leckekönyv listázó** osztályok, melyek elérik és tárolják az ott szükséges és létrehozott adatokat.



A tanárok a **Tanári kezelő felület** alatt érik el a saját funkcióikat, amik három csoportra bonthatók, **Üzenet küldő** (a diákoknak küldhető üzenetekre csoportosan és egyénileg is biztosít lehetőséget), **a Tantárgy kezelő** (itt tudják felvenni, módosítani, törölni az általuk tanított tárgyakat) és a **Tanuló értékelő** osztály (tantárgyakhoz rendelve tudják értékelni a diákokat).



A vezető a saját **Vezetői felületén** keresztül végezheti az adminisztratív feladatokat, az ide tartozó osztályok objektumai segítségével. Lehetősége van **elbírálni a beérkezett kérelmeket**, melyekről az érintettek üzenetben kapnak tájékoztatást, az **Üzenetküldő** segítségével a tanároknak és a diákoknak egyaránt küldhet üzenetet. Lehetősége van továbbá **kezelni a nyilvántartási** **adatokat** (az itt dolgozó és tanuló személyeket, valamint a tantárgyakat, bárminemű módosításról az érintettek üzenetben értesülnek), valamint egy **Statisztikakészítő** segítségével elérni az iskolai statisztikákat.

A fent részletezett modularizáció alapján elvégezve az osztályok csoportosítását, figyelembe véve azok függőségeit, a program eddig jól működőnek bizonyul.



1. A rendszer környezete

A tervezett rendszer egy kliens-szerver környezetben fog működni. A szerver az iskola területén fog elhelyezkedni, melyhez a felhasználók egy univerzális kliens felületen keresztül csatlakozhatnak.

A szervert a Microsoft Windows Server 2008 R2 operációs rendszere működteti, a kliens oldali elvárás legalább a Microsoft Windows XP SP3. A program .NET 3.5-ös verziójú keretrendszerben íródott, ezért ennek mind szerver, mind kliens oldalon telepítve kell lennie.



A rendszer környezete

1. A rendszer szerkezete

A tervezés során a szoftvert alrendszerekre bontottuk figyelembe véve a korábban meghatározott funkciókat és a modularizációt. Meghatároztuk azokat a külső elemeket és osztályszintű kapcsolatokat, amelyek szükségesek a szoftver implementálásához.



Alrendszerek és függőségeik

A szoftver felső, megjelenési szintjén található négy alrendszerbe lett csoportosítva, amelyek a felhasználói interfészek megjelenítéséhez szükséges osztályokat tartalmazzák figyelembe véve a különböző jogkörű felhasználók elvégezhető műveleteit.

A szoftver alkalmazási szintjén található alrendszerek a program működéséhez szükséges adatok tárolását, kezelését és hozzáférhetőségét biztosítják. A felső szinten lévő alrendszerek ezeket az itt lévő alrendszereket használják a működésükhöz.

Végül a köztes szinten a .NET 3.5-ös keretrendszerét szimbolizáló csomag áll.

Megjelenítési réteg:

A login alrendszer egy bejelentkező képernyőt foglal magában, melyben lehetőségünk van bejelentkezni a rendszerbe, szükség esetén új vezetőt tudunk regisztrálni.

A hallgató alrendszer a hallgató számára elérhető form és azok tab feladatait szemlélteti.

Hasonló szempontok szerint épül fel a vezetői és tanári alrendszer is, melyek az elvárt funkciók megvalósításának figyelembevételével szemlélteti a megjelenési réteg kapcsolódását az alkalmazásréteg funkcióihoz, úgy mint üzenetkezelés, tantárgykezelés, üzenetküldés és személyügyi adminisztráció.



A login alrendszer



A hallgató alrendszer



A Tanárok alrendszer



A Vezetői alrendszer

**Alkalmazási réteg:Állapot diagramok**

A programban jól elhatárolható állapotokat az alábbi osztályok, felületek vehetnek fel:

1. Megjelenítési réteg – Bejelentkező ablak



A bejelntkezőablak állapotai

A program elindítása után betölt a bejelentkező ablak, majd ebből az állapotból elindul egy automatikus keresés, ami ha talál már felvett vezetőt (létezik legalább egy vezető jogkörű felhasználó), átléptet az „Adatok várakozása” állapotra. Abban az esetben, ha nincs még regisztrált vezető letiltódik a felhasználónév és jelszó bevitel, csak egy gomb lesz aktív, amivel regisztrálhatjuk a legelső vezetőt. Bármilyen hiba esetén visszatérünk a „Login tiltása, kötelező vezető-regisztráció” állapotba, ellenkező esetben (sikeres volt a folyamat), továbblépünk a „Várakozás adatokra” állapothoz, melyben már nem látszódik regisztráló gomb és aktívak a beviteli mezők.

Hibás adatok megadásakor a felület nem enged tovább, érvényes felhasználónév és jelszó esetén továbbmegyünk a „helyes adatok állapotba”, melyből ismét automatikusan, lekérdezzük a bejelentkezett felhasználó jogkörét. Amint megjön az eredmény betöltjük a jogkörhöz tartozó formot és bezárjuk a bejelentkező ablakot.

1. Alkalmazás réteg – Üzenetkezelő osztály



Az üzenetek felvehető állapotai

Amint rálépünk a formok üzenetkezelő tabjaira elkezdjük betölteni a felhasználóhoz tartozó üzeneteket, amint elkészült és kiíródott a lista, belépünk az „Üzenetek listázva” állapotba, ahonnan két lehetőségünk nyílik. Amennyiben törölni akarunk egy levelet biztonsági okokból az adatbázisból nem törlődik, de a státusza töröltre módosul, így többet nem lesz listázva a felhasználónál. Egy új üzenet megnyitásakor a státusza olvasottá válik és vizuálisan el is különítődik a többi olvasatlan levéltől, úgy ahogy egy mindennapi levelezéskezelőtől ezt elvárhatjuk.

1. Alkalmazás réteg – Kérelemkezelő



Kérelmek kezelésének állapotai

Az elindított kérelmet először rögzítjük az adatbázisban, majd elküldjük a vezetőknek jóváhagyásra. Pozitív elbírálás esetén végrehatjuk a kérelemhez fűződő változtatásokat, majd jelezzük a döntés eredményét a kérelmezőnek. Elutasítás esetén szintén csak értesítjük a kérelmezőt.

1. Alkalmazás réteg – Hallgató osztály



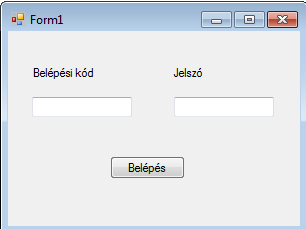
A hallgató lehetséges státuszai

Alapértelmezésként minden hallgató aktív státusszal indul, de kérelmezés útján elérhető számukra a kijelentkezett státusz, ekkor megszűnik a jogviszony, a hallgató nem lesz többé tagja az intézménynek, adatai archiválódnak. Élhetnek passzív félévvel, melyben a jogviszony tovább él, de tárgyakat nem vehet fel. Abban az esetben, ha minden tárgyat sikerül teljesítenie, a hallgató sikeresen elvégezte az iskolát.

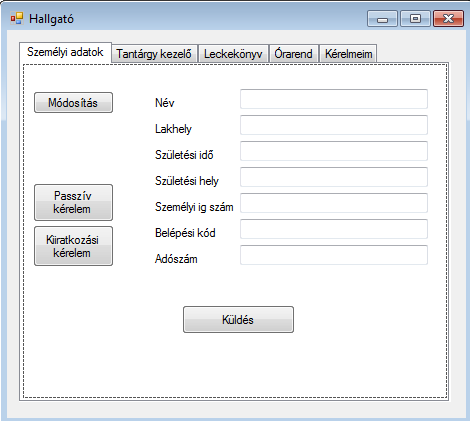
**Melléklet**

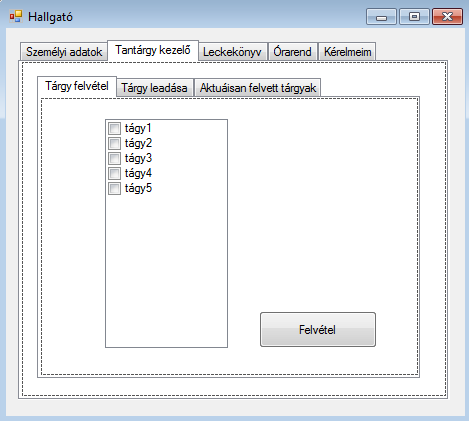
**Kezelőfelület látványtervek**

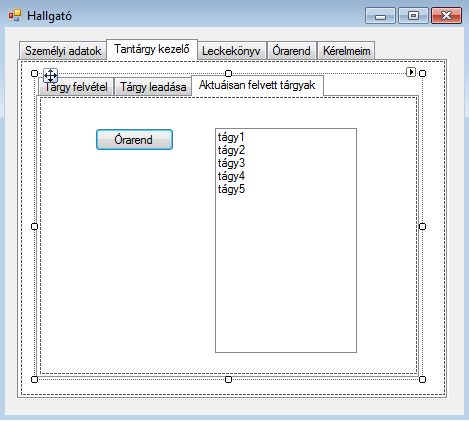
Bejelentkező ablak:

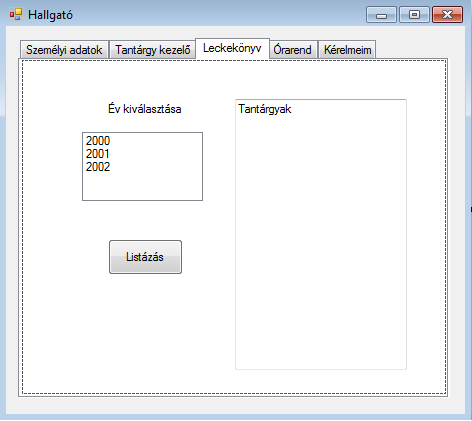


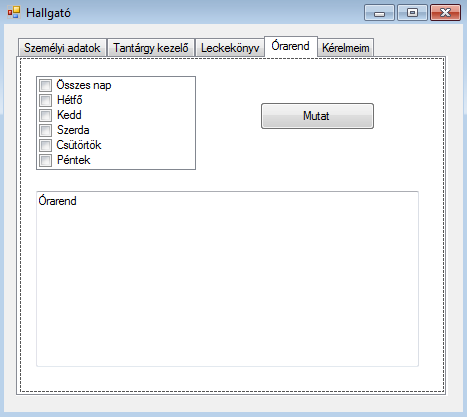
Hallgatói felület:

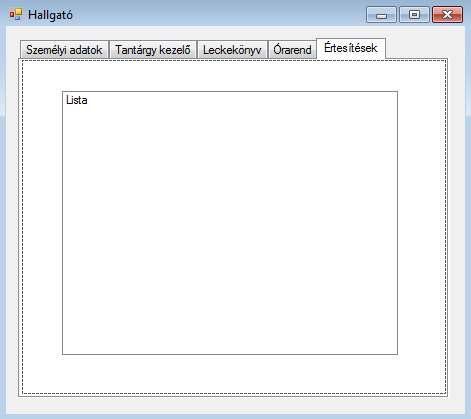
****

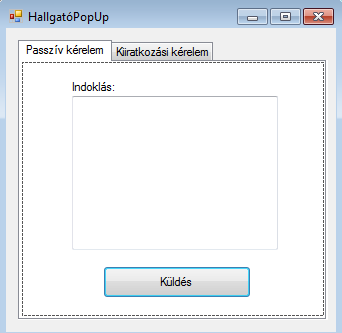
****

****

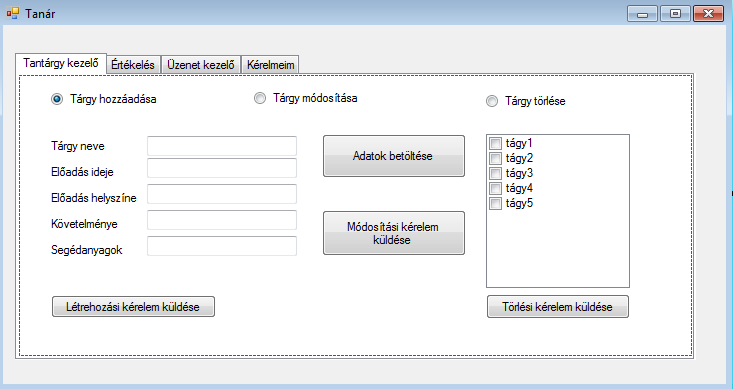
****

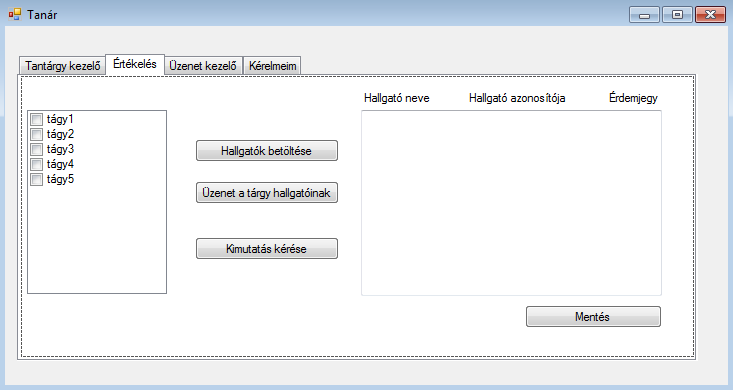
****

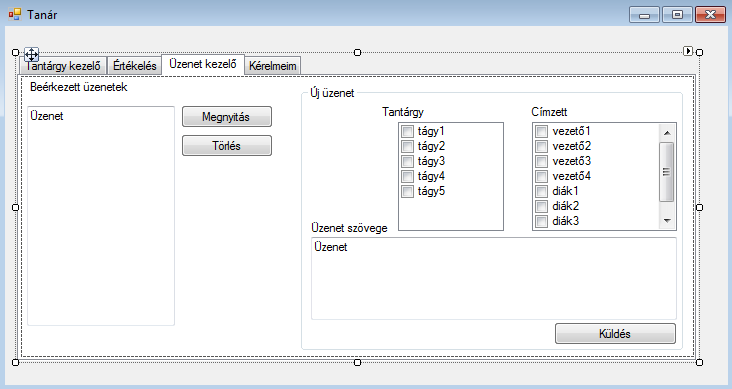
****

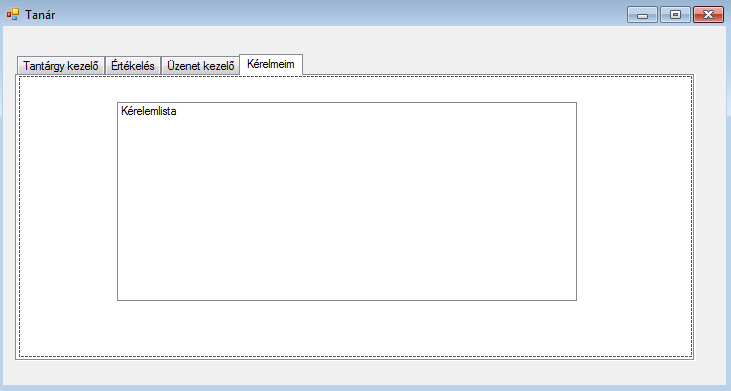
****

Tanári felület:

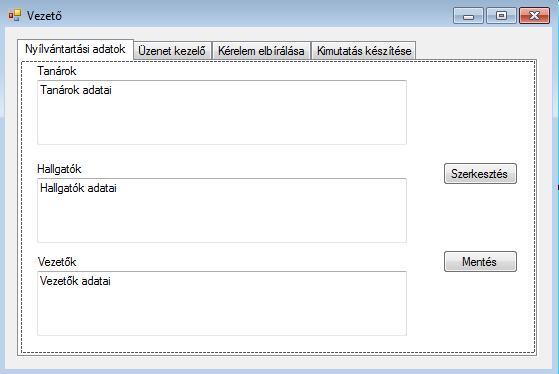
****

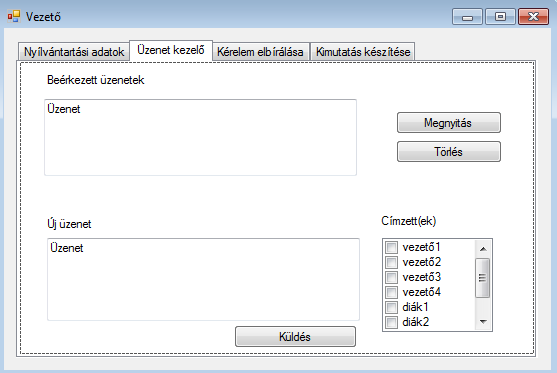
****

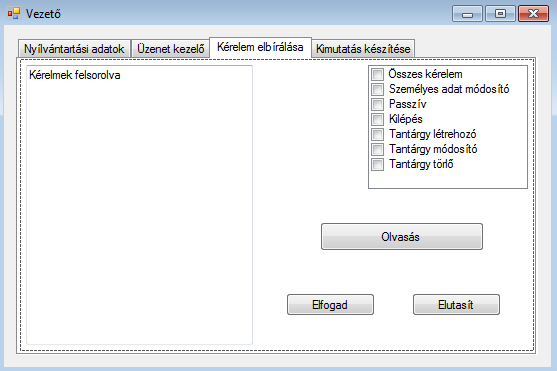
****

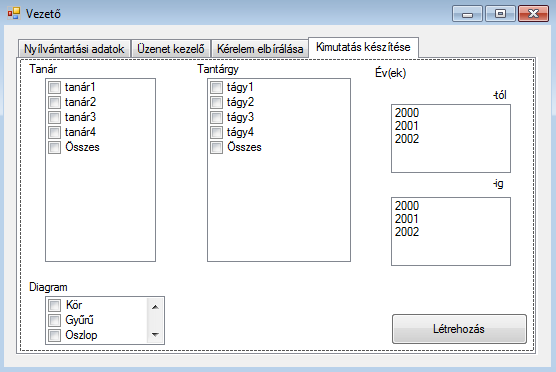
****

Vezetői felület:

****

****

****

****

**JEGYZŐKÖNYV**

**Időpont**: 2013. február 22.

**Jelen vannak:**

* Dori Bence
* Schönberger Zsófia
* Balogh Tamás
* Esse Zsolt

**Események:**

A projekt használati eset diagramjával kapcsolatos teendők megtárgyalása és szétosztása történt. A megegyezés szerint a csapat minden tagja elkészíti saját verzióját 2013. február 25.-én 20.00 óráig és feltölti a Github verziókövető verziókövetőre.

Esse Zsolt Dori Bence  
jegyzőkönyvvezető projektvezető

**JEGYZŐKÖNYV**

**Időpont**: 2013. március 1.

**Jelen vannak:**

* Dori Bence
* Schönberger Zsófia
* Balogh Tamás
* Esse Zsolt

**Események:**

A Szoftvertechnológia gyakorlat órán átismételtük az osztálydiagramokkal kapcsolatos tudnivalókat, ennek megfelelően szóban egyeztettük, hogy milyen teendők szükségesek a projekt során elkészítendő szoftver osztálydiagramjának létrehozására. Közösen elkészített vázlat alapján Balogh Tamás vállalta az osztálydiagram sztereotípiájának CASE programban történő megrajzolását. Dori Bence a használati eset diagramot módosítja a megbeszéltek alapján. Schönberger Zsófia és Esse Zsolt – szerepkörükből adódó kötelességeiken kívül a többi csapattag munkáit ellenőrzi, és szükség esetén javaslatokat tesznek vagy módosítják a már elkészült dokumentumokat (diagramokat).

Esse Zsolt Dori Bence  
jegyzőkönyvvezető projektvezető