

ESZTERHÁZY KÁROLY FŐISKOLA

MATEMATIKAI ÉS INFORMATIKAI INTÉZET

Botanikus kerti adatbázis 3D-s virtualizációja és webes megjelenítése

Készítette:

Szívós Attila

programtervező informatikus BSc

Szakál Tamás

programtervező informatikus BSc

Témavezető:

Dr. Kovács Emőd

egyetemi docens

Tartalomjegyzék

1. Vizualizáció	5
2. A projekt és a hallgatók kapcsolata	9
3. Unity 3D	12
3.1. Unity IDE	13
4. Botanikus rendszer	14
4.1. Pályázat	14
4.2. Az adatbázis	16
4.3. Alkalmazások	19
4.3.1. Unity kliens program	19
4.3.2. Adminisztrációs program	20
4.3.3. Frissítő program	22
4.3.4. Weboldal	22
4.4. Modellek	24
4.4.1. Blender	26
4.4.2. Unity Tree Creator	27
4.5. Fejlesztői környezet	28
5. Elért eredmények	29
6. A projekt jövője	31
6.1. Android alkalmazás	31
6.2. Egyéb fejlesztések	32

Bevezetés

A háromdimenziós vizualizáció területe igencsak szerteágazó és sokszínű. Vizualizáció alatt azt a folyamatot értjük, amikor egy megtervezett vagy már meglévő tárgyról valamilyen képet vagy akár egy háromdimenziós modellt alkotunk, számítógépes grafika segítségével. Rengeteg lehetőség áll előttünk a számítógépes vizualizáció területén, ezáltal sok mindenről tudunk modellt alkotni. A mi munkánk főként az előbb említett vizualizáció területén helyezkedik el, de ami mégis más, illetve egyedi a mi munkánkkal kapcsolatban az az, hogy a modell egy botanikus kert.

Megpróbáltunk egy olyan egyedi dolgot modellezni, mellyel, ha nem is csak mi, de kevesen foglalkoztak. Ezen vizualizáció fő célja a biológia oktatás kiegészítése, mellyel interaktív és vizuális oldalról közelíthetjük meg a tanulást. Több alkalmazás fejlesztése is fut jelenleg, melyeket az Eszterházy Károly Főiskola, főként biológia szakos hallgatói számára készítünk. A háromdimenziós vizualizáció és a hozzá tartozó egyéb fejlesztési feladatok egy projekt része, melynek fő célja az Eszterházy Károly Főiskolához tartozó botanikus kert növényvilágának és struktúrájának fejlesztése. A projekt kezdete óta rengeteg lehetőséget véltünk felfedezni ezen ötleten belül, melyeknek egy részét már sikeresen megvalósítottunk. A mostani mérföldkő a botanikus kert jelenlegi állapotát tükrözi, a háromdimenziós megjelenítő a mostani területet jeleníti meg egy virtuális valóságot hoz létre. A megújítási projekt befejezését követően az új állapotot is vizualizálni kell. Természetesen az előző állapot nem vész el, újra meg lehet nézni azt. Létrehoztunk egy professzionális adatbázisrendszert a botanikus kerti növények adatainak strukturált tárolásához, melyet könnyű menedzselni egy adminisztrációs program segítségével. A vizualizációhoz használt fejlesztői szoftvernek a Unity 3D-t választottuk. Ez a fejlesztői környezet multiplatformos játékok fejlesztésére szolgál és jelenleg nagyon elterjedt, rengetegen használják. Érdekes kombinációt értünk el ezzel, hiszen egy grafikus játékmotor végez el egy természeti terület vizualizációját. Ezzel megtartjuk az alkalmazás játék tulajdonságát, így játékos feladatokkal bővíthetjük annak szolgáltatási körét,

emellett természetesen szeretnénk az oktatási jellegét is megtartani. Munkánk egyrészt hasznos a Botanikus Kertet tanulmányozó kutatók és hallgatók számára, másrészt ez egy újszerű megközelítés a programtervező informatikusok számára, mely példát mutat arra, hogyan lehet egy játék fejlesztő eszközt használni az oktatásban.

Az új adatbázis és a menedzseléshez szükséges eszközök a Biológiai Tanszék munkatársainak készültek. Későbbiek során bemutatjuk, hogy az adatbázis tervezésénél és kialakításánál mennyire jól sikerült megfelelni a kutatók és hallgatók elvárásainak.

1. fejezet

Vizualizáció

A vizualizáció az informatikában valamilyen valós alak, forma, modellként való megjelenítést jelenti egy virtuális térben. Az első számítógépek még nem voltak képesek modellek megjelenítésére, csupán karaktereket, különböző írásjeleket voltak képesek feldolgozni. Azonban ahogy teltek az évek, úgy bővült az ismeretünk a számítástechnika területén, egyre kifinomultabb eszközöket és módszereket találtak fel, így mára már olyan szintre fejlődött a virtuális modellezés technológiája, hogy egy olyan komplex formát is, mint egy ember, képesek vagyunk szinte teljes részletességgel vizualizálni, lemodellezni.

A vizualizációt magába foglaló számítógépes grafika tudományának robbanásszerű fejlődését a szórakoztatóipar erre irányuló nagymértékű igénye váltotta ki. Ennek köszönhetően egyre több és szebb grafikával rendelkező játékot készítettek, illetve a filmbeli animációk mára már észrevehetetlenül beleolvadnak a valós térbe, de más területeken is teret hódított magának a számítógépes grafika csodája. A számítógépes grafikát általánosan több, egymástól elkülönülő ágakra bontjuk, melyeket céljuk szerint csoportosítunk:

1. Geometria: különböző felületek feldolgozásának és megjelenítésének tanulmánya
2. Animáció: mozgásábrázolás és megjelenítésének tanulmánya
3. Megjelenítés: különböző algoritmusokkal való képgenerálás tanulmánya
4. Képfeldolgozás: képi folyamatok és képszerkesztés tanulmánya
5. Topológia: terek és felületek viselkedésének tanulmánya.

Geometria: ezen terület a háromdimenziós objektumok virtuális térben való megjelenítését vizsgálja. Egy valós térbeli objektum virtuális térben való megjelenítését sok tényező

befolyásolhatja, ezek közül a legfontosabb a rendelkezésre álló információk, illetve a megjelenítéshez használt módszer. A legtöbb objektum jól ábrázolható kétdimenziós térben is, bár fennáll az objektum térbeli hatásának elvesztése. A leggyakrabban használt ábrázolási módszerek közé tartozik a hálós megjelenítés, melynek során a felületet poligonokra bontva jelenítik meg. Minél több poligonból áll egy felület, annál pontosabb, szebb képet kapunk eredményül.

Animáció: mikor egy felület, vagy más objektum mozgását, illetve deformálódását vizsgáljuk, akkor beszélünk animációról. A számítógépes grafika tudományágai közül ezen folyik a legtöbb fejlesztés, melynek eredményeképp ma már olyan filmeket nézhetünk, melyek teljes egészében számítógépes animációval készültek, illetve a játékok is rengeteg animációt tartalmaznak. Ezen kívül előfordulhat az élet bármely területén, mint a közoktatásban, vagy bármilyen tervezési folyamatnál.

Megjelenítés: megjelenítés (vagy renderelés) során egy modell két dimenzióba történő leképezése valósul meg. Két alapvető renderelési módszer létezik, az egyik a lokális illumináció, a másik pedig a globális illumináció. Lokális illumináció során egyszerűsödnek a fényviszonyok számításai, mivel csak a fényforrásból érkező fényt vesszük figyelembe, ezzel szemben a globális illumináció figyelembe vesz minden fényforrást és tükröződést, így egy valóság-hű, fizikailag helyes fényvisszaverődésekkel rendelkező modelltől kapunk képet, cserébe viszont a sokszorosára nő a számítási igény.

Képfeldolgozás: képfeldolgozásról beszélünk, mikor egy már meglévő kép átalakítását vagy elemzését végezzük. A kép átalakítása során egy újabb kép keletkezik, amely jobban megfelel céljainknak. Ez jelenthet zajcsökkentést, élesítést vagy akár teljes formai átalakítást is. Mikor egy képet elemzünk, annak információt igyekszünk kinyerni, mint a szintek vagy méretek értékei.

Topológia: a számítógépes grafika világában topológia alatt egy adott objektum drótvázának vizsgálatát, kezelését értjük. A drótvázak csúcsokból és élekből állnak, melyek kapcsolata oldalakat határoz meg. Minél több csúcsunk és élünk van, annál pontosabb, részletesebb drótvázát kapjuk a modellnek. A topológia azonban nem csupán egy állandó, mozgulatlan drótváz vizsgálatát értjük, hiszen egy objektum nem mindig fix. Animáció során egy objektum drótváza folyamatos változáson megy keresztül, ennek vizsgálata szintén a digitális topológia részét képezi.

A vizualizáció, mint grafikai folyamat nem sorolható egyértelműen egyik csoportba sem, mivel bár elsősorban geometriai ismeretek szükségesek egy objektum modellezéséhez, gyakran van szükség annak animációjára, illetve felületi drótvázának ismeretére is.

Vizualizáció során képezhetünk rajzokat, diagramokat, egyéb alakzatokat, bizonyos adatok felhasználásával. Az ilyen vizualizációk készülhetnek kétdimenziós vagy háromdimenziós kiterjedésben. A legelterjedtebb a kétdimenziós, hiszen egyszerűen ábrázolhatók vele rajzok, tervrajzok, mérési eredmények, kétdimenziós grafikonok ábrázolásai, mért adatok segítségével. A háromdimenziós vizualizációba tartozik például a modellezés valamilyen modellező szoftver segítségével. A háromdimenziós megjelenítés ismérve, hogy erőforrásigényesebb a felépítése és megjelenítése, de a mai számítógépek esetében ez már csak nagyon extrém esetben jelenthet gondot. A háromdimenziós komputergrafikában a háromdimenziós modellezés valamilyen objektum háromdimenziós felületének matematikai megjelenítésének folyamata. A létrejött dolgot nevezzük háromdimenziós modellnek. Ezeket a modelleket mi egy képernyőn két dimenzióban látjuk a háromdimenziós renderelés segítségével. Modelleket létrehozhatunk saját kezűleg vagy automatikusan is. A manuális modellezésnél mi magunk formázzuk a modellünket hasonlóan, mint a szobrászatban[1].

A természettudományos tárgyak tanítása esetén sokszor van olyan eset, amikor egy kísérlet szemléletesebbé tenné az adott problémát, viszont a kísérlet nem jöhet létre, mert

1. túl drága lenne az adott iskolának,
2. túl gyorsan, vagy túl lassan megy végbe ahhoz, hogy normálisan megvizsgáljuk,
3. túl veszélyes lenne a kísérletet végzők számára.

Ezért sokszor csak szimuláció útján hajtodik végre egy-egy kísérlet, melyet számítógép segítségével hoznak létre. A szimuláció mellett elegendő csak egy kísérlet vagy valamilyen tárgy, élőlény vizualizációja.

A munkánk során vizualizáltuk az Eszterházy Károly Főiskola botanikus kertjét, melyet egy háromdimenziós alkalmazáson keresztül tekinthetünk meg. A vizualizációs folyamat első lépése a környezet felmérése volt. Meglátogatva a botanikus kertet referenciapontokat vettünk fel a térképen és ezeken a referenciapontokon adott irányokba egy-egy fényképet csináltunk. A Unity fejlesztői környezetben ezek után a referencia-fényképeken keresztül modelleztük le a terület felületét. A felület létrehozása egy úgynevezett magasságtérkép segítségével is létrehozható. A magasságtérkép egy fekete-fehér térkép, melyen a legvilágo-

sabb rész lesz a legmagasabb pont, a legsötétebb pedig a legalacsonyabb. Ha rendelkezünk egy ilyen térképpel, akkor a Unity-be betöltve, a felület automatikusan generálódik. Egyéb objektumok is megtalálhatók a botanikus kertben, úgy mint táblák, kövek, kerítés, stb., ezek a Blender modellező szoftver segítségével készültek el.

2. fejezet

A projekt és a hallgatók kapcsolata

A botanikus kerti adatbázis és a hozzá tartozó szoftverek jó példát mutatnak egy komplex rendszer felépítésére és működésére. Az Eszterházy Károly Főiskolán a programtervező informatikus hallgatók iskolai éveiben kevés projekt van, azok sem túl nagyok. Ennek a rendszernek a bemutatása egy támpontot adhat a hallgatóknak, hogy miként is lehet egy ilyen összetett rendszert létrehozni. Bár egy nagy rendszer egészének a megvalósításakor több helyes út is létezik, könnyebb a hallgatóknak ezeket megmutatni, mint megvárni, hogy az ilyen dolgokra saját maguk jöjjenek rá. Az iskolai évek során rengeteg tapasztalatot lehet szerezni a fejlesztésekkel kapcsolatban, de sokat viszont csak akkor szereznek meg a hallgatók, amikor már elvégezték az iskolát és a munkába álltak.

A főiskolai oktatásban jelen lévő adatbázis készítés és menedzselési feladatok csak egy alapot szolgáltatnak ahhoz, hogy a hallgatók elmondhassák, hogy ők neki tudnak kezdeni egy nagy rendszer fejlesztéséhez. Azért írom azt, hogy alap, mert mi is csináltunk hasonló adatbázis fejlesztési feladatokat, de a botanikus kerti adminisztrációs program és a hozzá tartozó adatbázis sokkalta összetettebb, mint egy órai példa. Minél több táblából és kapcsolatból áll egy adatbázis úgy egyre hosszabbak és bonyolultabbak lehetnek a lekérdezések is. Ráadásul a botanikus kerti adatbázis rengeteg tárolt eljárást és függvényt tartalmaz, amelyek sokkal gyorsabbak és biztonságosabbak annál, mintha közvetlenül küldünk egy-egy adatmanipulációs parancsot a szervernek. Ez szintén egy jó példa, hiszen a programtervező informatikusok tanulnak PL/SQL nyelven tárolt eljárásokat írni, melyek a későbbiekben hasznukra válhatnak.

A teljes fejlesztési projekt az iskola keretein belül jött létre. Ez egy nagy lehetőség volt számunkra, amellyel éltünk is. Tapasztalataink és személyes véleményünk szerint minden

programozó hallgatónak egy fejlesztési projekten belül kellene megtapasztalnia azt, hogy milyen feladatokkal és természetesen milyen eredményekkel jár egy nagyobb projektben való részvétel. Tehát fontosnak tartjuk azt, ha egy programtervező informatikus hallgató az iskolai éveiben részt vegyen egy ilyen projektben. Vannak olyan tervezési és implementációs lépések, melyek abszolválásával kiszélesedik a gondolkodásmódjuk, jobban átláthatóvá válik előttük egy rendszer felépítésének lépcsőfokai és persze nem mellesleg megjelenik a tapasztalat. Mindenkit biztatunk arra, hogy ha lesz hasonló lehetősége az iskolai évek alatt, akkor habozás nélkül vágjon bele, mert csak hasznára fog válni.

A rendszer több részből áll, de az újszerűség és a technológia középpontjában a Unity áll. Mivel a főiskolán nem volt speciálisan Unity oktatás amikor tanultuk a programozás alapjait, így saját magunktól kezdtük el megismerni a rendszert. A Unity egy viszonylag új rendszer és az évek során egyre csak nőtt a népszerűsége. Példának kitűnő, hogy olyan dologgal foglalkozunk, ami új és fejlett. A projektmegbeszélések során célul lett kitűzve, hogy egy új és professzionális adatbázisrendszert alkossunk meg a főiskola botanikus kertjéhez. Rögtön adott volt a kapcsolódási pont a két rendszer között, a Unity alkalmazásnak ezt a rendszert kell használnia arra, hogy információkat szolgáltatson a felhasználóknak. Magától értetődő, hogy ha létezik egy adatbázis, akkor azt valamiképpen menedzselni kell és ekkor született meg egy adminisztrációs program tervezete. Ebből is látszik, hogy mennyire széles az informatika világa, hiszen rövid időn belül több, az informatika különböző területén elhelyezkedő technológiát kapcsolunk össze egy adatbázisra építve. Természetesen itt még nem álltunk meg, hiszen szükségszerű volt egy új weboldal készítése a botanikus kert számára, amely szintén a meglévő adatbázist használja. Ezen a gondolatmeneten továbbhaladva több érdekes dologgal is elő lehet állni, nem érdemes lekorlátozni arra, amink eddig van.

Egy új alkalmazási területtel bővítenénk a szolgáltatás palettáját, ami nem más, mint az Android operációs rendszer. A tervek között szerepel egy Android-os alkalmazás fejlesztése, ami szintén a botanikus adatbázishoz kapcsolódik. Egy ilyen kisebb projektet akár ki is lehetne adni főiskolai feladatnak, melyen egy vagy esetleg két ember dolgozik, ezzel is elősegítve azt, hogy a hallgatók belelássanak abba, hogy hogyan is kell aktívan részt vállalni egy éles rendszer fejlesztésében.

Nem szükséges ezt a projektet a főiskola keretein belül hagyni, akár már középiskolában is érdekes lehet egy ilyen rendszer megismerése. Nyilvánvalóan egy középiskolában nem kell annyira részletesen belelátni a működésbe, mint a főiskolán, de egy realisztikus három-

dimenziós alkalmazás talán sokkal több tanuló képzeletét fogná meg egy középiskolában. Tehát érdemes lenne kiszélesíteni ezt a látásmódot, emellett pedig nem szabad megfeledkezni a rendszer tartalmáról sem.

Egy komplett növény-, állatvilág és természeti terület került modellezésre, mely főként a biológia szakos hallgatók számára érdekes. Ezzel egyik célunk a természettudományi tárgyak oktatásának fejlesztése az informatikán keresztül. Ami eddig létrejött, azt saját magunk terveztük meg, de az első körben segítséget kértünk néhány biológia szakos hallgatótól is, miként vélekednek a háromdimenziós alkalmazásról és a természettudományi tárgyak tanulásáról. A későbbiekben szívesen veszünk minden új ötletet a hallgatóktól, hogy mivel lehet még szórakoztatóbbá és hasznosabbá tenni az alkalmazást.

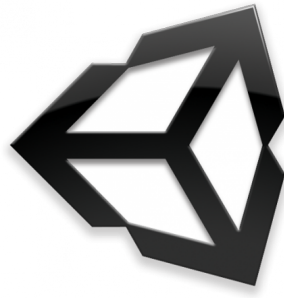
A jelenlegi rendszer funkcióival és a jövőbeni ötletek rendszerbe való beépítésével megpróbáljuk a mostani és a leendő tanárok módszertanának fellendítését, kiegészítését. Nyilván a mi munkánk témája a természettudományi tárgyakon belül leginkább a biológiával foglalkozik, de el lehet képzelni, hogy miként lehet hasonló alkalmazásokkal más területeken is ösztönözni a tanulni vágyókat. Ebben ez esetben nem feltétlenül arra kell gondolni, hogy szintén egy adatbázisra épülő rendszert kellene fejleszteni, hiszen minden terület más és más dolgokkal foglalkozik. Például az informatikán keresztül a Unity segítségével akár fizikai vagy természeti jelenségeket is le lehetne szimulálni. Ha nem csak a főiskolán belül gondolkozunk ilyen lehetőségeken, akkor a fiatalabb korosztályoknak is sokkal több játékos megoldás fejleszthető tanítási céllal. Ha jobban belegondolunk, ez elég jó megoldás, hiszen a Unity-t eleve játékok fejlesztésére találták ki, így ezáltal a fejlesztéskor az adott korosztálynak megfelelő játékos és interaktív szintet lehetne kifejleszteni.

A fejlesztések során több biológus szakos hallgatóval is felvettük a kapcsolatot és kikértük a véleményüket, hogy mit is gondolnak a háromdimenziós megjelenítésről. Az ötlet mindenkinek tetszett és örömmel nézték meg a bemutatónkat. Az alkalmazások tesztelésére az augusztus hónapban kerül sor, mely során megkérjük az eddig megismert hallgatókat és más ismerősöket, hogy teszteljék a rendszert. Tervbe van véve egy kérdőív létrehozása, melynek egy része azzal foglalkozik, hogy mindenki leírhatja, mit is változtatna a programon. Ezeket az ötleteket figyelembe véve a későbbiekben úgy próbáljuk megformálni az alkalmazást, hogy a lehető legtöbb hallgatói igénynek megfeleljen.

3. fejezet

Unity 3D

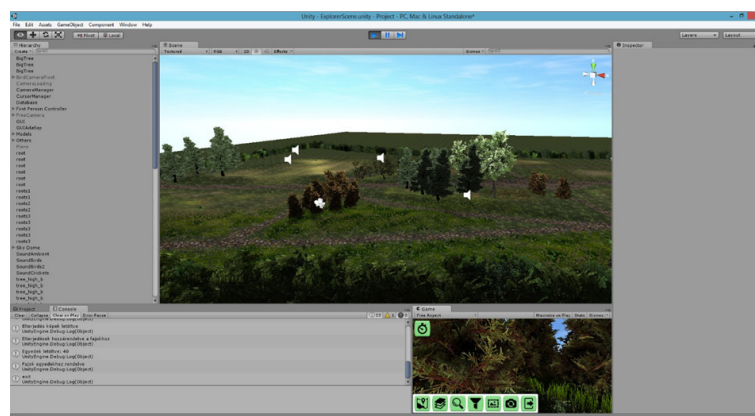
A Unity Game Engine-t a san francisco-i székhelyű Unity Technologies cég fejlesztette ki 2005-ben. A Unity nagyon gyorsan betört a játékfejlesztés piacára, hiszen sokkal egyszerűbb a használata, mint a hagyományos keretrendszereknek. Rengeteg előre megírt egység található a rendszerben, mely felgyorsítja a fejlesztés folyamatát. A Unity-t egy összegben vagy havidíjas formában megvásárolva is használhatjuk. 2009-ben kiadták a rendszer ingyenes változatát, melyben néhány dolgot nem használhatunk. Választhatunk a C#, JavaScript vagy Boo nyelvek között a fejlesztés során. A rendszer magja C++ nyelven íródott, de a fejlesztés más nyelven történik. Az úgynevezett szkripteket akár az előbb felsorolt mindhárom nyelven is megírhatjuk. Egy projekt többféle platformra is lefordítható egyetlen kattintással, például iOS, Android, Windows, Linux vagy akár konzolgépekre is. Használhatunk akár DirectX 11-et és Shader Model 5.0-t is. Ezen kívül rengeteg technológiát használ a Unity, melyeket a létező játékmotorok is használnak, úgy mint például Bump Mapping, Dynamic Shadow vagy akár Screen Space Ambient Occlusion. Ezzel a rendszerrel elég nagy versenyhelyzet alakult ki a játékfejlesztés területén. Ha egy kisebb játékfejlesztő csapat vagy egy baráti társaság arra adja a fejét, hogy játékot szeretnének fejleszteni, akkor a Unity egy nagyon jó választás. A Unity-t használó eszközökről érdekes és friss statisztikákat lehet találni a Unity hivatalos oldalán[2]. Mivel a Unity játékok fejlesztésére készült, ezért azt gondoljuk, hogy egyedülálló vállalkozásba kezdtünk, hogy egy oktatást segítő alkalmazást hoztunk létre.



3.1. ábra. A Unity logója

3.1. Unity IDE

A Unity 3D-val való fejlesztés során az úgynevezett Unity Editor-t használjuk arra, hogy felépítsük a háromdimenziós környezetet. Ez az integrált fejlesztői környezet tökéletes arra, hogy néhány egérekattintás segítségével létrehozzunk egy működő alkalmazást. Természetesen szükséges az alkalmazás tartalmát importálni, tehát a hangokat, 3D-s modelleket, textúrákat, stb. Az úgynevezett jelenetek felépítése még nem elegendő, hiszen a programok logikáját is szükséges megírni, hogy ne csak egy statikus alkalmazást kapjunk. Ehhez használhatunk külső fejlesztői eszközöket úgy, mint a Visual Studio, vagy a Unity 3D telepítési csomagjában található MonoDevelop, de akár egy egyszerű szövegszerkesztővel is megírhatjuk kódjainkat. A Unity Editor modulárisan felépülő szoftver, több ablakból áll. Külön ablak van a háromdimenziós környezetben való navigáláshoz, a játék futtatására, objektumok tulajdonságának megjelenítésére. Ez egy nagyon jól megkonstruált fejlesztői környezet, átlátható és egyértelmű. Az alkalmazások futtathatók az Editor-on belül is vagy pedig a fordítás után közvetlenül futtatva az alkalmazás indítófájlját.



3.2. ábra. A Unity Editor

4. fejezet

Botanikus rendszer

4.1. Pályázat

Az Eszterházy Károly Főiskola Botanikus kertje egy sokak által látogatott, gazdag élővilággal rendelkező, csaknem 10 000 négyzetméter nagyságú terület. Mivel nagyon közel van a főiskola Almagyar dombi épületeihez, így a hallgatók számára is egy gyakran látogatott helyé vált. A kert 1967-ben került a főiskola tulajdonába. Megközelítőleg 600 növényfajnak ad otthont, melyek közül 200 védett. A gyűjtemény helyi jelentőségét két fő értéke adja:

1. bemutatja a hazai flóra valamint a Kárpát-medence növényfajait
2. növeli a fennmaradási esélyét azoknak a védett növényfajoknak és társulásoknak melyek a Bükk-hegységből, valamint az Északi-középhegység más élőhelyeiről származnak

Az egyre növekvő látogatottság, és a növényvilág folyamatos bővülése egy nagyszabású fejlesztés kivitelezését sürgette, melyet egy pályázat megnyerésével sikerült is megkezdni. Ennek során megújítják a magyarországi tájegységek növényvilágát bemutató élőhelyeket, valamint mesterséges körülmények között szaporítanak több ritka, a Bükk-hegységben előforduló növényfajt. A beruházás során a főiskola olyan egyedi kertszerkezetet kíván létrehozni, ami egyrészt figyelembe veszi a kert jelenlegi adottságait (elhelyezkedés, meglévő növénytársulások stb.), másrészt közvetett eredményként jól szolgálhatja a főiskola és Eger város egyéb kapcsolódó tevékenységeit is. A legfontosabb programelemek és elérendő rész-célok:

1. a nem vizes élőhelyek, természetközeli társulások rekonstrukciója, fás és lágyszárú növényanyag pótlása, kert élőhelyfoltjainak fejlesztése
2. speciális élőhelyekhez, gyűjteményekhez kapcsolódó beruházások
3. vizes élőhely, lápteknő rekonstruálása, illetve kialakítása
4. védett növényfajok ex situ megőrzési programjához kapcsolódóan egy fajtól függő stabil állomány létrehozása és fenntartása, amely alkalmas az adott faj genetikai állományának fenntartására (lehetőség szerint a genetikai leromlást elkerülve), illetve egy törzsállomány létrehozása 40-45 taxonból

Mindezek mellett fejlesztés alatt van egy alkalmazás, melynek célja a botanikus kert teljes vizualizációja, így a látogatók anélkül képesek bejárni a kertet, hogy ténylegesen ott lennének. Továbbá készült egy adminisztrációs alkalmazás is, ami a kert adatbázisát hivatott karbantartani, illetve a botanikus kert weboldala is megújul, alkalmazkodva az új igényekhez. Ezen alkalmazások fejlesztése a mi feladatunk, melynek során folyamatosan tartottuk a kapcsolatot Pénzesné dr. Kónya Erikával, aki megbízott minket ezen alkalmazások fejlesztésével.



4.1. ábra. A Botanikus kert

4.2. Az adatbázis

Az első feladatunk az volt, hogy megtervezzünk és felépítsünk egy olyan adatbázist, amely a botanikus kerti növények adatait fogja tárolni. Legfőbb szempont az volt, hogy ingyenes adatbázis-kezelő rendszert használjunk. Így esett a választás az Oracle MySQL adatbázis-rendszerére. A fejlesztés korai szakaszában, tehát a tesztelési időszak előtt, helyi adatbázisrendszert használtunk. Ehhez a XAMPP csomagot választottuk, mely már tartalmaz egy MySQL adatbázisszerver alkalmazást. A folyamatos adatbázis módosításokat egy közös tárhelyen osztottuk meg egymással, hogy mindig a lehető legújabb változatot használjuk. Az adatbázis jelenlegi struktúrája következetes és a jövőbeni fejlesztésekre előre fel lett készítve. A növények adatait több csoportra oszthatjuk:

1. azok az adatok, melyek teljesen nyilvánosak, mindenki számára elérhetők,
2. vannak korlátozottan nyilvános adatok, ezek a törzsadatok, csak bizonyos jogosultsági szint felett érhetők el,
3. vannak privát adatok, melyeket csak magas jogkörrel rendelkező személyek tekinthetnek meg.

Teljesen nyilvános adatoknak tekintjük a botanikus kertben fellelhető növények angol, magyar, latin neveit, maximális magasságát, védettségi szintjét, vertikális elhelyezkedését. Természetesen egy bizonyos jogszint alatt a védett vagy fokozottan védett növények nem jelennek meg a felhasználó számára.

A törzsadatok azok az adatok, melyek a növények további tulajdonságait írják le. Ezeknek az adatoknak az eléréséhez már nagyobb jogkörrel kell rendelkeznie a felhasználónak. Itt olyan tulajdonságokat tárolunk a növényekről, mint például a levél széle, levél formája, szár típusa, termés típusa, stb.

A privát adatokat kizárólag a kutatók és adminisztrátorok láthatják. Ide azok az adatok kerülnek, melyek a növényvilág további bővítéséhez szükséges információkat tárolják, illetve a bizalmas kutatási anyagok is itt kerülnek feljegyzésre.

Ezt az adatbázist közösen használják az alkalmazások. Ugyanazzal a regisztrált profillal tudnak a felhasználók bejelentkezni a weboldalra, a háromdimenziós alkalmazásba és az adminisztrációs programba is. Mivel az adminisztrációs alkalmazást nem használhatja mindenki, a bejelentkezéskor egy speciális jog szükséges.

A felhasználók csoportokba kerülnek. Amikor valaki végigcsinálja a regisztrációs folyamatot, akkor a létrejövő profil automatikusan bekerül a vendég csoportba. A csoportokat jellemezzük jogokkal, így ha egy felhasználó bekerül egy csoportba, akkor örökli annak jogait. Egy felhasználó több csoport tagja is lehet egy időben, ilyenkor a közös jogokon túl a másik csoport jogait is megkapja. Alapértelmezetten több, előre definiált csoport létezik az adatbázisban:

1. vendég
2. hallgató
3. szerkesztő
4. oktató
5. kutató
6. adminisztrátor
7. fejlesztő

Az adminisztrációs programon belül lehetőség van új csoportok létrehozására vagy épp a jelenlegiek módosítására, törlésére. Természetesen csoportot csak akkor lehet törölni, ha nincs egyetlen tagja sem.

A növényfajok tárolásához viszonylag sok adat megadása szükséges. Egyrészt meg kell adni az alapadatokat, melyek mindenki számára nyilvánosak, ezen túl a korlátozottan nyilvános és a privát adatok megadása nélkül nem tárolható egy faj. Amikor egy faj bekerül az adatbázisba, akkor abból a fajból készíthetünk egyedet. Egyed felvételénél hivatkozunk a felvenni kívánt egyed fajára, majd pedig egyéb egyedspecifikus tulajdonságokat kell megadni. Az egyedek rendelkeznek hely információval és saját egyedgalériájuk is van. Az egyedgalériákon túl vannak eseménygalériák, melyben az egyes botanikus kerti események képeit tároljuk. Közvetlenül a képeket nem az adatbázisban tároljuk, hanem csak egy elérési útvonalat mentünk el, mely a szerveren az adott képre mutat.

A fajokról, egyedekről és felhasználókról további információkat is elmentünk, ezeket a bejegyzéseket a napló táblában találjuk. A naplóban tároljuk, hogy ki és mikor vette fel a bejegyzést és, hogy mi a bejegyzés szövege. Többféle naplózás létezik a rendszeren belül:

1. egyedállapot naplózás
2. irodalom naplózás
3. kutatási naplózás
4. rendszernaplózás

Az egyedállapot naplózásán azokat a hasznos információkat értjük, melyek egy-egy egyedhez kapcsolódnak. Ezeket az információk a korlátozottan nyilvános kategóriába sorolhatjuk, tehát a vendégeken kívül mindenki látja. Ilyen naplóbejegyzés lehet például, ha valamilyen aktuális státuszt jegyzünk fel egy egyeddel kapcsolatban.

Az irodalom naplózás egy-egy fajhoz köthető, szintén korlátozottan nyilvános adat. Ez az információ főként az oktatóknak és hallgatóknak hasznos, hiszen az oktatók feltölthetnek ide akár könyvek címeit is, mely tartalmazza azt, hogy az adott fajról hol lehet olvasni további hasznos információkat.

A kutatási naplózás egy magasabb rendű naplózási forma. A naplóbejegyzés többnyire nem függ semmitől, tehát nem az egyedek állapotairól hordoz információt. Kutatások alatt létrejött bejegyzéseket tárolunk ebben a formában és nyilván ezeket a bejegyzéseket csak kutató vagy magasabb rangú felhasználó írhatja és olvashatja.

Létezik egy teljesen elszeparált naplózási forma a naplótípusok között és ez pedig a rendszernaplózás. Az adminisztrációs program folyamatainál a szerver megjegyzi, hogy melyik felhasználó mit csinált a rendszerben. Ezáltal, ha valaki valamilyen hibát vét, akkor ez visszakereshető a rendszernaplóból. Ez a fajta naplózás csak az adminisztrációs programnál fontos, hiszen azon keresztül történik az adatbázis kezelése.

Egy verzió táblában tároljuk a programok aktuális verziószámát és az előző verzióhoz képest történt módosításokat. A programok a futás elején megvizsgálják, hogy a táblában lévő verziószám megegyezik-e a sajátjukkal, és ha találnak újabbat, akkor letöltik a legújabb verziókat a frissítő program segítségével.

Ezen adatbázis-szerkezet többszöri újragondolás után valósult meg, így végül egy rendkívül kiforrott, komplex és stabil rendszert sikerült elkészíteni az alkalmazások alá. Gondolva a jövőbeli fejlesztésekre, helyeztünk bele több olyan funkciót is, melyek kihasználtsága még nem teljes értékű, azonban amint a fejlesztés eljut arra a szintre, mikor már nagy szükség lenne a bővítésre, már nem kell komoly változtatásokat eszközölni a rendszerben.

4.3. Alkalmazások

Több alkalmazás is készült a botanikus szolgáltatási csomagon belül, melyeket az Eszterházy Károly Főiskola hallgatói és oktatói, valamint kutatói és vezetői számára fejlesztettünk. A csomag tartalmazza a Unity 3D megjelenítőt, az adminisztrációs és a frissítő alkalmazásokat, valamint az új weboldal is része.

4.3.1. Unity kliens program

A Unity 3D-s alkalmazáson keresztül nyerhetünk betekintést az Eszterházy Károly Főiskola Botanikus kertjének csodálatos növény- és állatvilágába. Az alkalmazás három fő részből áll.

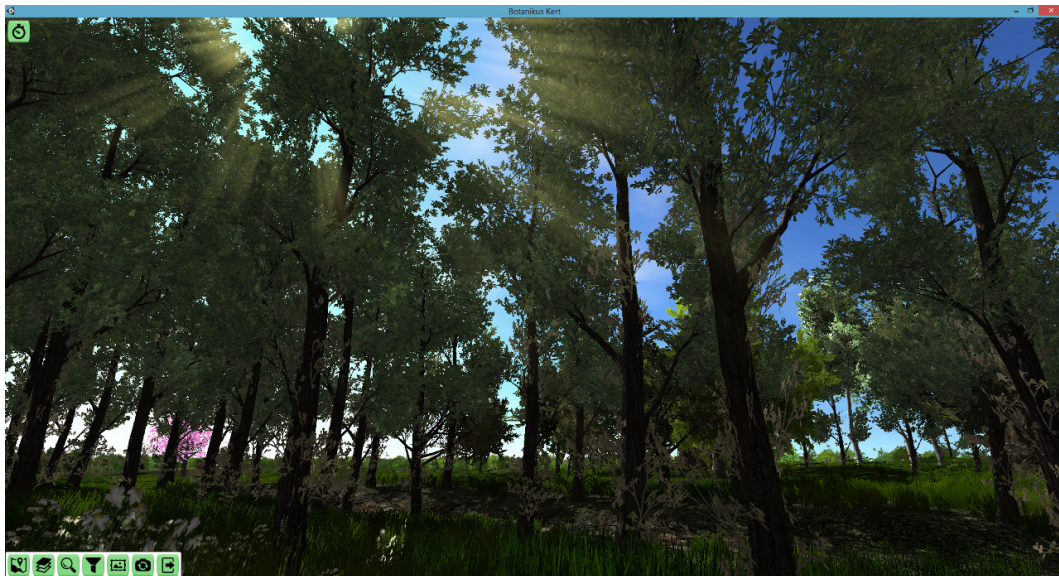
Első lépésben a program egy bejelentkezést követel meg. Enélkül nem lehet használni a program többi részét. A bejelentkezéskor megadott név és jelszó párossal a weboldalra is be lehet jelentkezni. A megfelelő név és jelszó megadása után a rendszer betölti a kétdimenziós részt, ami a második nagyobb része az alkalmazásnak.

A kétdimenziós térkép egy gyors és átfogó betekintést ad a botanikus kerti növények helyzetéről és további információkat találunk ezekről a növényekről. Minden növényt egy egyedi szimbolikus jellel jelölünk a térképen. Ez a szimbolikus jel fajonként egyedi. Ezzel a módszerrel nagyjából meg lehet különböztetni a fajokat a térképen. Egy adott egyedre kattintva a térképen, megjelenik annak adatlapja. Ezen az adatlapon nem feltétlenül jelenik meg minden információ. Ha vendégként lépünk be, akkor viszonylag kevés, míg kutatóként viszont nagyon sok adatot látunk az egyedekről. Ez egyfajta biztonságot jelent, hiszen léteznek az adatbázisban nem nyilvános adatok is, melyek megtekintését nem mindenkinek engedünk meg. A bal alsó sarokban megjelenő menüsorban több hasznos funkciót is találunk. Alapfunkciók a szűrés, keresés, vertikális szűrés, továbbá a galériák és növényadatlapok megjelenítése.

A harmadik nagyobb egység, mely a fő erőssége az alkalmazásnak, az a háromdimenziós megjelenítés. Többször ellátogattunk a botanikus kertbe, ahol bizonyos referenciapontokon fényképeket készítettünk, hogy a virtuális terep megalkotása ezen referencia fényképek alapján történjen. A megtervezésben a botanikus kert növényei is segítettek, de ezeket csak a felület vizualizációjára használtuk fel, hiszen a növények valójában a botanikus kerti adatbázisból kerülnek be a programba. Tehát a kétdimenziós térképen kattintva, miután ki-

választottuk a menüsorból a nézetváltást, a háromdimenziós jelenet töltődik be. Itt szabadon tudunk mozogni az iránygombok és az egér segítségével úgy, mint egy számítógépes játékban. A realisztikus hatást növeli a repülő apró rovarok, madarak és kisebb állatok, illetve a napszakok váltakozása, természetbeli hangok, fák szél általi mozgásai, stb. Ebben a nézetben is ugyanazok a funkciók jelennek meg, melyeket a kétdimenziós térképes nézetben találunk, néhány dologgal kiegészítve.

Felmerül a kérdés, hogy milyen teljesítményű számítógépen érdemes és lehet futtatni a klienst. A Unity keretrendszer játékok fejlesztésére lett kitalálva, így a maximális vizuális hatás eléréséhez egy nagyobb teljesítményű számítógépre lesz szükségünk. Természetesen a grafikai beállításokat át lehet állítani a számítógépünk teljesítményének megfelelően, hogy a gyengébb teljesítményű gépeken is viszonylag gyorsan fusson. Ilyenkor több dolgot ki-kapcsolhat a kliens, hogy a lehető leggyorsabban fusson az alkalmazás. Több optimalizációs eljárás is bekerült a programba. Az egyik ilyen az úgynevezett LOD (Level of Detail) automatikus változtatása. Ez azt eredményezi, hogy minél távolabb vagyunk egy objektumtól, annál jobban csökken a modell részletessége.



4.2. ábra. A 3D nézet

4.3.2. Adminisztrációs program

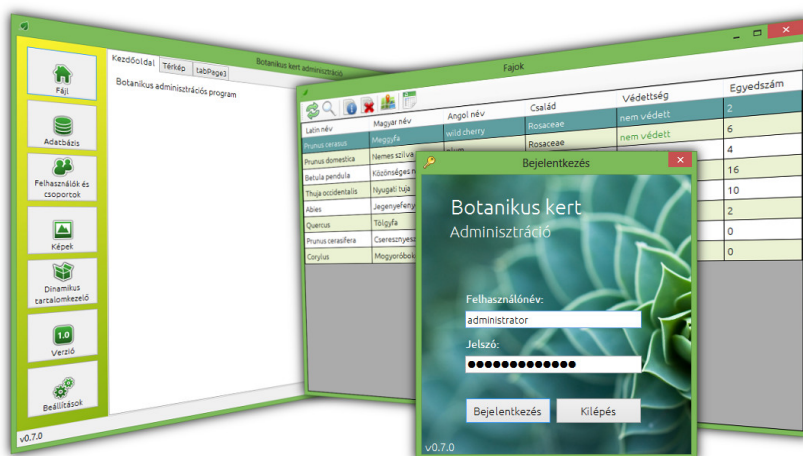
A botanikus kerti adatbázisban történő adatok menedzseléséhez szükségünk van az adminisztrációs programra. Ez a program C# nyelven íródott Windows Form alkalmazás. Ennek elindítását követően be kell jelentkezni a rendszerbe, hasonlóan a Unity alkalmazáshoz.

Mindkét helyen ugyanazokat a bejelentkezési adatokat használhatjuk. Az adminisztrációs programba csak akkor jelentkezhet be egy felhasználó, ha rendelkezik a bejelentkezéshez szükséges joggal.

Az adminisztrációs felület bal oldalán elhelyezkedő menüoszlop tartalmazza a menedzseléshez szükséges menüpontokat. Az Adatbázis menüben tudunk új fajt létrehozni és ebből az újonnan létrehozott fajból ezután már egyedeket is lehet készíteni. Ebben a menüben megtalálható még a Törzsadatkezelő funkció, melynek segítségével a fajokra jellemző tulajdonságokat lehet menedzselni.

Külön tudjuk kezelni a felhasználókat és a csoportokat a megfelelő menüpontok segítségével. A csoportkezelőben tudjuk a jogokat is kezelni, hiszen a jogok a csoportokat jellemzik, nem a felhasználókat. A felhasználó a csoportjától kapja meg a jogokat. Például egy kutató szintű felhasználó beléphet az adminisztrációs alkalmazásba, de a Felhasználók és csoportok menüt nem tudja előhívni.

Megtalálható továbbá egy Verziókezelő menü, melyben a feltöltött adminisztrációs és 3D alkalmazás verzióit lehet kezelni. Amikor egy frissebb verzió kerül feltöltésre, a kliens programok induláskor ellenőrzik ezt és egy frissítő alkalmazás segítségével frissülnek az alkalmazások. Ez azért szükséges, mert mint minden programban, itt is előfordulhatnak hibák, új funkció bevezetésére van szükség vagy éppen az adatbázis módosítások miatt a programokban is követni kell ezeket a módosításokat és a frissített programokat el kell juttatni a felhasználókhoz. A frissítés folyamata automatikusan végbemegy létező internetkapcsolat esetén.



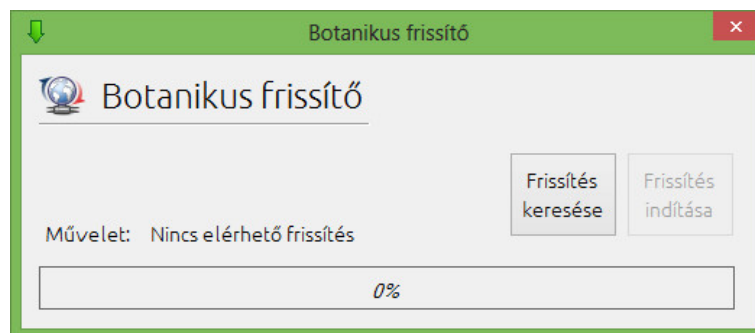
4.3. ábra. Az adminisztrációs program

4.3.3. Frissítő program

A korábban említett frissítési folyamatért egy frissítő alkalmazás a felelős. Ez a program szintén C# nyelven írt Windows Form alkalmazás. Egy program nem tudja saját magát frissíteni, hiszen nem tudja a fájljait felülírni futás közben. Erre a feladatra egy különálló program áll rendelkezésre, mely felülírja az alkalmazás régi fájljait.

A két kliens program induláskor rögtön ellenőrzi, hogy létezik-e frissebb verzió, és ha igen, akkor automatikusan elindítja a frissítő alkalmazást, amely frissíti a klienseket. A frissítések fontos szerepet játszanak abban, hogy a felhasználók mindig a legújabb verziójú programokat használják. Egy-egy hiba kijavítására vagy új funkció bevezetésére, módosítására új verzió készül el, melyet az adminisztrációs programon belül a Verziókezelőben tudunk feltölteni abban az esetben, ha rendelkezünk megfelelő joggal.

A frissítés során egy folyamatjelző csík mutatja a frissítés aktuális állapotát. Ha a frissítés idő előtt megszakad valamilyen oknál fogva, akkor használhatjuk továbbra is a régi verziójú alkalmazásokat, mert amíg a frissítő le nem tölt minden új fájlt, addig a régiék nem kerülnek felülírásra. Manuálisan is el tudjuk indítani a frissítő alkalmazást, de az általános felhasználóknak erre nincs szükségük. Egyszerűen elindítják valamelyik kliens alkalmazást, ami ellenőrzi az új verzió jelenlétét az adatbázisban, és frissítés esetén bezárja a kliens programot, majd elindítja a frissítő alkalmazást. Ilyen esetben a frissítő alkalmazáson belül nincs szükség interakcióra, automatikus elindul a frissítés folyamata.



4.4. ábra. A frissítő alkalmazás

4.3.4. Weboldal

A fenti alkalmazások számára biztosítani kellett egy felületet, ahonnan azokat a felhasználók könnyedén elérhetik, miközben tájékozódnak a botanikus kerttel kapcsolatos érdekességekről.

ról, eseményekről. Mindezek figyelembe vételével létrehoztunk egy weboldalt, ahonnan regisztráció után elérhetővé válik az alkalmazások letöltése, illetve a megfelelő jogosultságok mellett igénybe vehetőek az oldal további funkciói.

Az oldal tartalmának elérhetősége több jogcsoportra van bontva, melyeket dinamikusan kezelünk, minden csoportnak egyedi jogai vannak, ezek függvényében változik az oldal tartalmának elérhetősége is. A weboldal alapját egy CodeIgniter[3] névre hallgató webfejlesztő rendszer adja. Ez egy PHP alapú, nyílt forráskódú keretrendszer, mely az MVC[4] tervezési mintát támogatja.

Az MVC tervezési minta egy olyan programozási sémát határoz meg, melyben a programozó különválasztja a programkód adatfeldolgozásért és üzleti logikáért felelős részét a felhasználói nézettől, majd ezeket egymástól különálló rétegekbe csoportosítja. Az adatbázis műveleteket végző programkódok a modell (model) rétegbe, a felhasználói felületet leíró programkódok a nézet (view) rétegbe kerülnek, majd ezeket egy harmadik réteggel, a vezérlő (controller) réteggel kapcsolja össze. Ez a réteg gondoskodik arról, hogy a modell és nézet rétegek képesek legyenek megfelelően együtt működni. Ezen tervezési minta használatával képesek vagyunk elérni azt, hogy a felhasználói felület ne befolyásolja az adatkezelést, ezáltal az adatok átszervezhetőek legyenek a felhasználói felület változtatása nélkül, illetve az egyes modellekhez egyszerre több, különböző nézetet is társíthatunk, az igényeknek megfelelően.

Az oldalon való navigálást egy horizontálisan elhelyezett menüsor segíti, ahonnan elérhetőek a weboldal főbb funkciói, mint a növények részletes leírását tartalmazó menüpont, a botanikus kerttel kapcsolatos információk, illetve hírek és a weboldal karbantartásához szükséges menüpontok.

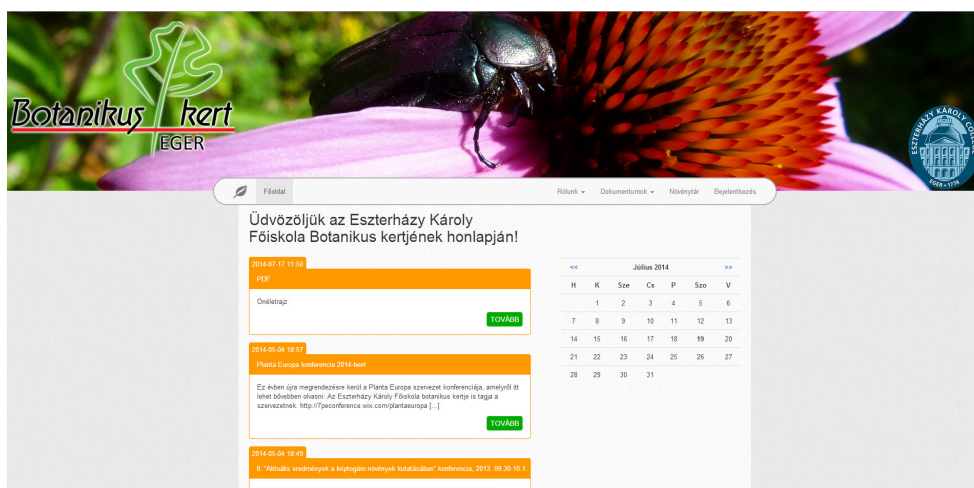
A weboldalra való regisztráció során a felhasználó megkapja a legalacsonyabb szintű jogokkal rendelkező csoport jogait, vendégként kerül nyilvántartásba, mellyel elérhetővé válnak számára az oldal fontosabb pontjai, a háromdimenziós alkalmazás letöltése, illetve a Növénytár menüpont. Magasabb szintű jogcsoportba kerülés az adminisztrációs alkalmazáson keresztül kerül kiosztásra, az adminisztrátor által.

A Főoldalról nyomon követhetőek a botanikus kerttel kapcsolatos legfrissebb hírek, illetve események. Maga a főoldal két részre bontható. A bal oldalon jelennek meg a hírek, bejelentések, melyeket az adminisztrátor, vagy egy erre kijelölt felhasználó hozhat létre, mó-

dosíthat, illetve törölhet. A jobb oldalon pedig az eseménynaptár lett elhelyezve, mely tartalmazza a kerttel kapcsolatos elmúlt, illetve jövőbeli események listáját.

A hírek kezelésére csak a megfelelő jogkörrel rendelkező felhasználók jogosultak. Ennek a menüpontnak az elérése szintén a horizontális menüből érhető el, itt van lehetőség a felhasználónak hírek írására, módosítására, vagy akár a törlésére is. A hírek egy egy oszlopos táblázatban jelennek meg, melynek bal oldalán a hír címe, jobb oldalán pedig a hírhez tartozó módosítás és törlés ikonok helyezkednek el. Új hír létrehozását a hírek listája feletti gombon keresztül lehet elérni.

A Növénytár a menü azon része, ahol a botanikus kertben élő növényfajok listáját van lehetőség megtekinteni, néhány alapvető adattal együtt. Ezen felül, ha egy felhasználó rendelkezik az ehhez szükséges jogokkal, megtekintheti az egyes fajokhoz tartozó egyedek teljes listáját, azok részletes adataival. Ha van az egyedről készült kép az adatbázisban, akkor azokat is itt lehet elérni.



4.5. ábra. A weboldal

4.4. Modellek

A Unity alkalmazásba betöltött növényeket meg is kell jeleníteni. A megjelenítés alatt modellek kirajzolását kell érteni. Többféle modell szerepel a háromdimenziós jelenetben, melyek nem egyfajta programmal készültek. A tereptárgyak, úgy, mint a faház és táblák, a Blender modellező szoftverrel készültek. A növényeknél többféle lehetőséget is szemügyre vettünk. A cél az volt, hogy ingyenes program vagy programok segítségével készítsük el a növényeket. Sok növény generáló program létezik, melyeket ki is próbáltunk. Például a tree[d],

Forester Lite és Arbaro. Ezeknél nagyon egyszerűen lehet legenerálni egy fát vagy bokrot, csak a paramétereket kell állítani, hogy a kívánt eredmény megszülessen. A Unity Editorban is van egy fa generáló modul. Végül is erre esett a választás.

Ennél az eszköznél már kicsit több paramétert kell beállítani, hogy realisztikus fa generálódjon, de meg van az az előnye, hogy a Wind Zone objektummal reakcióba lépjen, tehát a fák mozoghatnak, és úgy látszódnak, mintha a szél mozgatná azokat. Természetesen nagyon oda kell figyelni, hogy a részletes fáknál ne vigyük túlzásba az ágak és levelek számát, hiszen akkor nagyon sok lapból fog állni egy fa. Törekedni kell arra, hogy minél kevesebb lapból álljon, de még realisztikus maradjon.

A növénymodellek elkészítése a folyamatos adatbázis bővítéssel párhuzamosan működik. Ha az adatbázis bővítésekor új faj kerül felvitelre, akkor el kell készíteni hozzá a modellt. A modell úgy kerül a kliensekhez, hogy a Unity kliens letölti a modellesomagot, majd pedig betölti a futó programba. A modelleket egybe lehet fordítani a programmal, viszont ha új modellre van szükség, akkor az egész programot frissíteni kell. Ellenben a másik megoldás, ahogyan működik most a kliens, az az, hogy mindig letölti a legfrissebb modellesomagot. Ezáltal ki lehet váltani az egész program cseréjét, ha új modellre van szükség, egyszerűen csak létrehozuk az új modellt, belefördítjük a modellesomagba és a kliens legközelebb már a bővített csomagot használja.



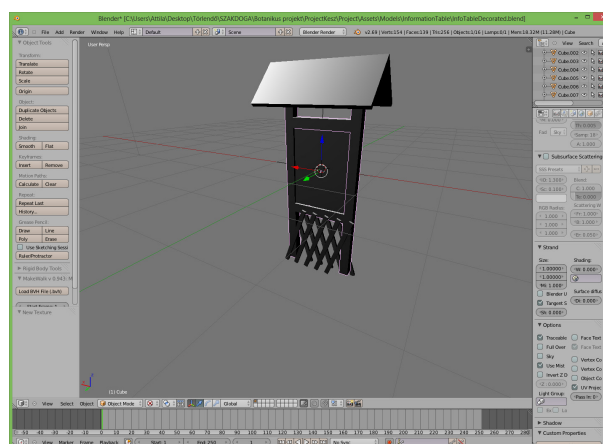
4.6. ábra. Modellek

4.4.1. Blender

A Blender[5] egy 3D modellező szoftver, mellyel 3D modelleket, animációkat készíthetünk. A botanikus kertben sok tereptárgyat le kellett modellezni, hogy a legélethűbb vizualizációját kapjuk. Az ilyen tárgyak modellezésére a Blender-t választottuk. A modelleket három verzióban készítettük el.

Eredetileg a Level of Detail sémát James H. Clark javasolta 1976-ban és a mai napig módosítás nélkül ezt használja a legtöbb 3D-s alkalmazás. Ez a módszer úgy működik, hogy minden objektumról több modell verziót készítünk, melyek közt csak a részletesség az egyetlen különbség. Futásidőben a megfelelő részletességű modell fogja reprezentálni az objektumot. Amíg a távoli objektumok kevésbé részletes modellt használnak, addig a poligonszám kisebb lesz és ezáltal megnövekedik a renderelési sebesség. Mivel a modelleket előre elkészítjük, a modellek részletessége egységesen csökken minden oldalon és emiatt sokszor nevezzük a diszkrét LOD-ot izotropikus vagy látószögfüggetlen LOD-nak[6].

Ezt a módszert követve növeltük a Unity 3D alkalmazás teljesítményét. A mi esetünkben is igaz az, hogy ha egy fa modell távol van, akkor nem szükséges a legrészletesebbet megjeleníteni, mert abban a távolságban a kisebb részletességű modell is ugyanúgy néz ki. Ilyen esetekben több lépcsőfokot is lehet alkalmazni, de a mi esetünkben létezik egy részletes objektum, egy közepesen részletes és egy nagyon alacsony poligonszámú modell. Tehát a tereptárgyakról és növényekről több modellt is készítettünk. A botanikus kertben fellelhető tárgyak például a sziklakert, kerítés, faház, táblák, stb. Szintén elkészíthetők Blenderben azok a növények, melyek nem kreálhatók a Unity beépített fakészítőjével.

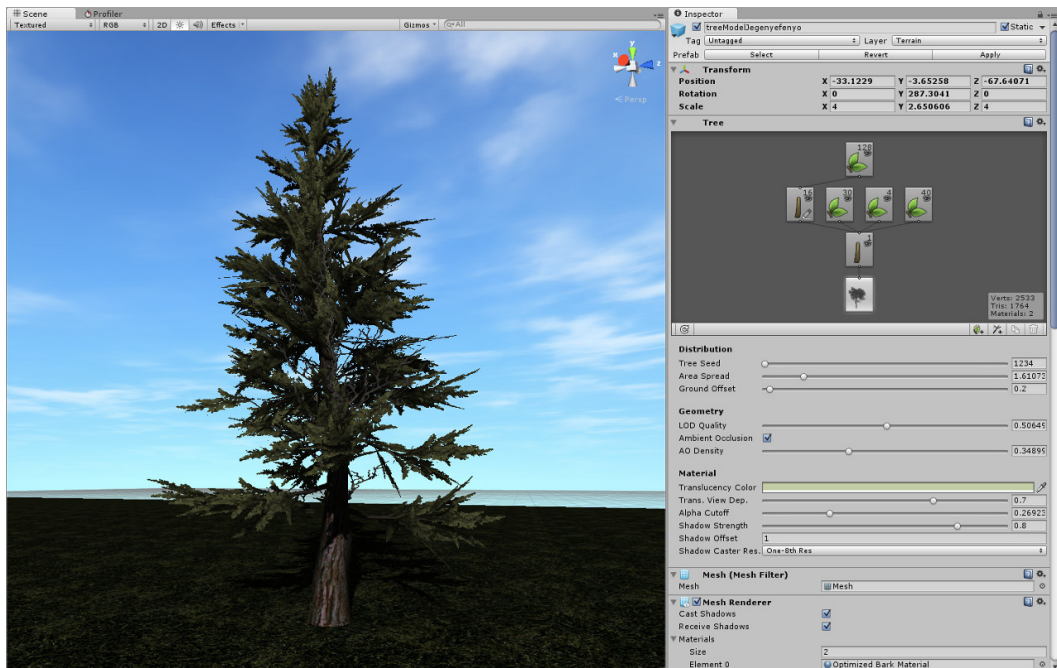


4.7. ábra. A Blender felülete

4.4.2. Unity Tree Creator

A Unity-nek van egy beépített fakészítő eszköze. Ezzel az eszközzel nagyon gyorsan lehet valóság-hű fa modelleket készíteni. Több oka is van, hogy miért ezt az eszközt használtam a famodellek elkészítésekor.

1. a modellek elkészítése nagyon egyszerű, én határozom a modell struktúráját egy fa szerkezetben, majd pedig legenerálódik a modell
2. valóság-hű fákat lehet készíteni viszonylag rövid idő alatt
3. a fák modellezése egyszerű, nem kell UV textúrázást csinálni
4. ezek a modellek reagálnak a Unity szélgenerátor objektumával, ezáltal mozognak a fák, külső modellekkel nem lehet ilyet csinálni, hacsak nem a modellezőszoftverben le nem animáljuk a mozgást, ami igen időigényes
5. manuálisan lehet állítani a LOD szintjét (bár itt nem lehet nagyon levenni ezt a szintet) az ágakra



4.8. ábra. A Unity Tree Creator modul

4.5. Fejlesztői környezet

A fejlesztések során több eszközt is felhasználtunk. A fejlesztés kezdeti szakaszában helyi adatbázist használtunk a saját gépünkön, később a főiskolán belül kaptunk egy szervert, amelyen megtalálhatók azok a szerverszolgáltatások, amelyekre szükségünk van. Tehát MySQL adatbázisszerver, FTP szerver, webszerver.

A háromdimenziós alkalmazáshoz a Unity Editor-t használtuk, amelyről már korábban szó volt. Ez egy nagyon egyszerűen használható fejlesztői eszköz. A Unity Editor mellett szükség volt egy olyan eszközre, amellyel a Unity-n belül tudunk írni úgynevezett szkripteket a 3D alkalmazáshoz. Ehhez a Visual Studio 2013-at használtuk, amely segítségével az adminisztrációs és a frissítő alkalmazást is fejlesztettük.

A weboldal fejlesztése eleinte helyi hálózaton történt a XAMPP szoftvercsomag segítségével, amely tartalmazta a számunkra szükséges Apache webszerver és MySQL adatbázis rendszer szolgáltatásokat, emellett támogatta a PHP nyelv legújabb verzióját, melyben a teljes fejlesztés történt. Később a főiskola által kínált szervert használtuk, így a fejlesztés a továbbiakban már egy ténylegesen, távolról elérhető szerverről folyt. Fejlesztői környezetnek a PHP Designer-t választottuk, ez egy PHP editor, mely weboldalak programozásához nyújt segítséget. Segítségével akár egyszerre is programozhatunk PHP, HTML és SQL nyelveken.

A teljes weboldal magját PHP[7] nyelven írtuk, ez egy szerver oldali szkript nyelv, melyet dinamikus weblapok megalkotására hoztak létre. A PHP-ban írt kódrészleteket a webszerver dolgozza fel, tehát az így írt kód ténylegesen nem látható a weboldal forráskódjában, csak annak kimenete.

5. fejezet

Elért eredmények

Az eddigi kutatások és fejlesztések során rengeteg érdekes ötlettel és lehetőséggel találkoztunk, melyeket igyekszünk lépésről lépésre megvalósítani. A fejlesztések ezek után sem állnak le, folyamatos karbantartást és további fejlesztéseket próbálunk kivitelezni. A 3D alkalmazás jelenlegi állapota egy olyan stabil verzió, mely az alap funkciók nagyobb részét már tartalmazza és a növények megjelenítése folyamatosan bővül, ahogyan készülnek a modellek és feltöltik az adatbázist adatokkal.

A weboldal fejlesztése is érdekes része volt a projektnek, hiszen nem csak egy kellően biztonságos és stabil oldalt kellett létrehozni, hanem nagyon fontos volt, hogy olyan felhasználói környezetet valósítsunk meg, mely elég letisztult és egyértelmű ahhoz, hogy bármely, az oldalra látogató felhasználó számára ne okozzon nehézséget az oldalon való navigáció. Ennek érdekében folyamatosan tartottuk a kapcsolatot a projekt vezetőjével, illetve több hallgatóval is konzultáltunk az oldal megjelenésével, illetve funkcióinak bővítésével kapcsolatban. Eredményül egy stabil, biztonságos és izgalmas oldalt sikerült létrehozni, amely alkalmazza a kor rendelkezésre álló legújabb és legbiztonságosabb fejlesztő eszközeit és módszereit, emellett képes kiszolgálni a felhasználók igényeit egy letisztult, igényes felületen keresztül.

A hallgatókkal együtt kooperálva sikerült egy egyszerűen használható és látványos 3D alkalmazást létrehozni és reméljük, hogy minél többen fogják használni ezt a rendszert. Reméljük sikerül megvalósítani azt a rengeteg lehetőséget, amiket ez az ötlet rejt még magában. Szeretnénk továbbra is foglalkozni ezekkel a szolgáltatásokkal és a továbbiakban is fejleszteni, de a projekt vezetője is azt szeretné, ha ezek után is mi foglalkoznánk a rendszer kar-

bantartásával. Ez azért is jó dolog, mert nagyon érdekes téma és reméljük, hogy a rendszer tökéletesítésével kimondhatjuk, hogy egyedi dolgot készítettünk.

6. fejezet

A projekt jövője

6.1. Android alkalmazás

A tervek között szerepel egy olyan okostelefonos alkalmazás fejlesztése, amellyel képesek lehetünk a botanikus kertet járva felismerni annak növényeit, illetve a térkép segítségével lokalizálni az egyes növények helyét.

Tartalmazna egy kereső funkciót is, melynek segítségével globálisan kereshetünk a növények bármely tulajdonságára, megtekinthető lenne az adott faj teljes leírása, illetve az egyes egyedekhez tartozó összes kép, mindez egy egyszerű alkalmazáson keresztül.

Ezen alkalmazás nagyon hasznos lehet az oktatás során is, hiszen segítségével a kertben járó hallgatók gyorsan és egyszerűen kaphatnak információt a keresett, vagy az éppen előttük lévő növényről. Emellett a kertben megforduló látogatók, túra csoportok is nagy hasznát vehetik, hiszen a térkép segítségével pontosan meg tudják határozni az általuk látogatni kívánt növény helyét, és a bejárható útvonalak irányát.

Minden növénynél el lehetne helyezni egy NFC tag-et vagy QR kódot és a telefont a tag-hez odatartva vagy pedig a telefon kamerájával felismertette a QR-kódot, rögtön megjelenik az adott növény adatlapja, így kiesik a keresés folyamata. Egyre jobban terjed az RFID technológia és már a nagyobb városokban is találunk például szobrok előtt NFC tag-et, melyhez odaérintve a telefont, azonnal információkkal szolgálja ki a felhasználót az adott szoborral kapcsolatban.

6.2. Egyéb fejlesztések

Ahogy egyre többen használják a programot, úgy egyre több igény merül fel a használhatósággal kapcsolatban. Tervezett módon is kikérdezzük a hallgatókat, hogy az alapfunkciókon kívül milyen egyéb funkciót szeretnének viszontlátni az alkalmazásban.

Mivel játékos dolgokat is bele szeretnénk vinni a Unity alkalmazásba, így több dolog is eszünkbe jutott, melyek közül néhányat érdemes lenne megvalósítani. Lehetne valamilyen kérdéseket kapcsolni az adott növényekhez, elrejtett növényeket megkeresni a botanikus kertben vagy akár egy virtuális túrát szervezni, ahol szerver-kliens modellt kellene kialakítani és több felhasználó barangolhat együtt a botanikus kertben. Ezen túrák szervezésére a weboldalon is lenne lehetőség, ahol a túra ideje és útvonala megjelenik a botanikus kerti alkalmazás egyszerűsített térképén, esetleg az túrára való jelentkezés is itt valósulna meg.

Ha kész lesz a növényvilág teljes vizualizációja, akkor a botanikus kertben jelen lévő kisebb állatokat is vizualizálnánk. Mivel a kertben sok lepke fordul elő, így ezeket lemodelleznénk és mozognának is. Vagy éppen madarak röpködnének a fák fölött. Tervezett fejlesztések között szerepel a botanikus kert különböző évszakokban való megjelenítése is, így a felhasználók figyelemmel kísérhetik a botanikus kert növény- és állatvilágának különböző évszakok közötti váltakozását is. Ezek mind olyan fejlesztési lehetőségek, amik egyrészt sokkal részletesebbé varázsolnák az alkalmazást, másrészt pedig vonzóbbá tehetnék a hallgatók körében a program használatát.

Pályamunka lezárásának dátuma: 2014.07.20.

Irodalomjegyzék

- [1] STUART K.CARD, JOCK D. MACKINLAY, BEN SHNEIDERMAN: Reading in Information Visualization: Using Vision to Think, Morgan Kaufmann, 1999
- [2] UNITY HARDWARE STATISTICS: <http://stats.unity3d.com/index.html>
- [3] CODEIGNITER: <http://ellislab.com/codeigniter/user-guide>
- [4] MVC DESIGN PATTERN: <http://www.phpro.org/tutorials/Model-View-Controller-MVC.html>
- [5] BLENDER: <http://www.blender.org>
- [6] DAVID P. LUEBKE: Level of Detail for 3D Graphics, Morgan Kaufmann, 2003
- [7] PHP: <http://hu1.php.net/manual/en>