

ORACLE MULTIMEDIA RENDSZERTERV

ÖNÁLLÓ LABOR
8. FÉLÉV

KÉSZÍTETTE: ANKUCZA TAMÁS, BEVMTO

2008. MÁJUS 3.

1. FUNKCIONÁLIS TERV

1.1. Feladat részletes specifikációja

A feladat egy multimédiás adatbázis, és a hozzá tartozó mobiltelefonokra optimalizált web portál elkészítése. A felhasználók, akik mobiltelefonjuk web böngészőjével megnyitják a portált, regisztrálhatnak. Regisztráció után lehetőségük lesz saját multimédiás tartalom feltöltésére. Ilyen multimédiás tartalom elsősorban videó, illetve kép fájlok. Multimédiás fájljaikat kategorizálhatják, úgynevezett galériákba rendezhetik. Megtekinthetik, sőt ezen kívül osztályozhatják (tetszési indexszel láthatják el) valamint kommentárt fűzhetnek mások feltöltött tartalmához. Ezek a kategóriák előre definiáltak, ezzel is megkönnyítve a multimédiás tartalmak közötti tallózást, keresgélést. Adott kategóriákon belül aetszési indexek átlaga alapján toplista készül. Minden felhasználónak lesznek „kedvencei”. A kedvencek közé bárki által készített bármilyen tartalmat hozzáadhat. Ez is egyféle népszerűségi mutató, hiszen meg lehet állapítani, melyek azok a fájlok, amelyeket a legtöbbnek neveztek kedvencüknek. A portálon lehetségesen lesz „nyereményjáték” is, különböző meghirdetett témákra, címekre lehet nevezni saját videóval, fotóval. Ezeket (is) a többi felhasználó értékelése (osztályzata) alapján rangsoroljuk. A rendszer több felhasználós, vagyis egyszerre több felhasználó folytathat kommunikációt a szerverrel. Még részletesebben a következő pontokban fejtem ki a specifikációt.

1.2. A rendszer célja, motivációja, helye a világban

A rendszer tervezése a BME-VIK (hagyományos, 5 éves képzéses) mérnök-informatikus szak 8. félévének tantervében szereplő önálló labor keretein belül történik. Mint ahogy a neve is mutatja (önálló), a hallgatónak a feladata önállóan elkészíteni a félév során a feladatát, konzulensei iránymutatása által. Ezt az önálló munkát értékelik (osztályozzák) mind a konzulensek, mind a tárgyfelelős. Mivel egy ember (hallgató) munkabírása is véges, ezért egy félév alatt kevesebb várható el tőle, mint egy teljes csapatától, amit példának okáért egy vállalat az adott projektre létrehozott. Így a környezet miatt a feladat lényegesen egyszerűsített, sokkal kevésbé komplex, mint a való élet hasonló projektjei. Sok funkció elhagyásra kerül, csak a lényeges pontok kerülnek előtérbe. Ettől eltekintve a feladat egy egész (és így értékelhető) projekt. A tervezendő rendszer nem a nagyvilág számára készül

elsősorban, tehát az elhanyagoltabb részek közé fog tartozni a munkatervben is már említett tesztelési terv, és megvalósítása (különböző hibatűréseket csak felületesen fejtem ki), valamint az üzemeltetési terv, hiszen ebből a rendszerből nem lesz „éles” rendszer.

A feladatot egy kicsit részletesebben specifikálva, a tervezett adatbázis illetve portál elsősorban a sport, mint fogalom köré fog épülni. A portál a különböző sportágak rajongóinak igényeit szolgálja, teszi még izgalmasabbá a mindennapokat. A korábban említett kategóriák elsősorban az egyes sportágakhoz lesznek köthetőek. Másodsorban pedig a multimédiás tartalmak címkézésének köszönhetően lesznek több sportágat átölelő kategóriák is (például „legszebb pillanatok” vagy akár „olimpiai döntő”, stb.). Ezekre a címkékre természetesen lehet keresni is, a könnyebb navigációt elősegítendő.

1.3. Az egyes szereplők igényei, céljai, szempontjai

A rendszert egy adott pillanatban két féle felhasználó rangú csoport használhatja. Egyik csoport az, aki „csak” böngészik, tartalmakat néz, míg a másik csoport, akik éppen tartalmat (legyen akár video, akár fotó) töltenek fel a portálra. Mivel a felhasználók mobiltelefonjaikon keresztül használhatják a rendszert, így akármerre is járnak a világban, az internet segítségével bárhol tölthetnek fel tartalmat, akár olyat is, amelyeket épp akkor készítettek/rögzítettek például egy sporteseményen. A már említett „nyereményjáték” is kapcsolódhat ehhez, például „Ki készíti a legjobb pillanatfelvételt a Labdarúgó Európa bajnokság során?”. Ugyanakkor a mobilitásnak köszönhetően bármilyen helyzetben tudják használni a szolgáltatást, képesek a videók illetve a fotók között böngészni akár utazás közbeni időtöltés gyanánt. Így a portál, illetve annak használata remélhetőleg (legalábbis ha a nagyközönség számára is elérhető lesz, vagyis túlmutatva az önálló labor keretein) a sportkedvelő közönség mindennapos tevékenységévé válik. De persze, mivel ehhez megfelelő marketing szükségeltetik (és ez sem része az önálló labori tevékenységemnek), ez egyelőre csak egy terv a lehetséges jövőre vonatkozóan. Ám ez a mobilitás hátrányokkal is jár. Ehhez a szolgáltatáshoz (a nagymennyiségű adatátvitel miatt) szükség van gyors elérésre. A gyors relatív fogalom, ám biztosak lehetünk abban, hogy egy felhasználó nem szeretne hosszú perceket várni, mire

fel/letöltődik az adott multimédiás tartalom. Ezt kívánt sebességet jelenleg csak 3G kapcsolattal lehet elérni, így tehát a szolgáltatás jelenleg korlátozódik olyan helyszínekre, ahol elérhető a 3G. Persze ez nem kizáró ok, a felhasználó dönthet úgy, hogy „várakozik”.

A rendszerben a következő típusú emberek használhatják:

1.3.1. Böngésző felhasználó

Az a felhasználói csoport, aki az adott pillanatban a rendszer felhasználói felületének a „böngésző” részét látja/használja. Multimédiás tartalmakat nézeget, esetleg véleményeket (comment-eket) ír hozzájuk vagy értékeli azokat a tetszési index-el. Böngészés közben hozzáadhatja az adott fájlt a kedvencei közé is. Jogköreiben ez a felhasználói csoport azonos a feltöltő felhasználóval, csakis az adott pillanatbeli cselekvésük dönti el, hogy hova tartoznak.

1.3.2. Feltöltő felhasználó

Az a felhasználói csoport, aki az adott pillanatban a rendszer felhasználói felületének a „feltöltési” részét látja/használja. Saját eszköze (mobiltelefon elsősorban) memóriájáról (beépített, vagy memóriakártya) multimédiás tartalmat tölt fel a rendszerbe. Nevet ad neki, címkéket ragaszt rá, kategóriába sorolja, esetleg jelzi, hogy ezzel a tartalommal indulni kíván valamelyik éppen aktuális nyereményjátékon. Jogköreiben ez a felhasználói csoport azonos a böngésző felhasználóval, csakis az adott pillanatbeli cselekvésük dönti el, hogy hova tartoznak.

1.3.3. Moderátor

Az a felhasználói csoport, akinek a jogosultsága annyival bővül az előzőekben említetteken kívül (tehát ő is mondható böngészőnek illetve feltöltőnek), hogy a feltöltött tartalmakat, vagy az azokhoz fűzött kommentárokat moderálhatja. Tehát ha a feltöltött tartalom sérti valamelyik törvényt és/vagy a portál szabályait, akkor törölheti azt, illetve ha valamelyik felhasználó kommentje nem odaillő (erről részletesen a portál szabályzata fog nyilatkozni), akkor módosíthatja/törölheti azt. Szélsőséges esetben kitilthat felhasználókat (a kitiltott felhasználó értelem szerűen nem használhatja a rendszert).

1.3.4. Szerver oldal

A szerver oldalon nem beszélhetünk igazi felhasználóról. A fejlesztők (ez esetben egyedül én, mint önálló laboros hallgató) elkészítik a rendszert, majd ezen túl már csak a megrendelői igények alapján kell rajta módosítani (mivel felhasználói oldalról a moderátor felfogható „felhasználói adminisztrátornak” is), ám ez már nem része az önálló labornak.

1.4. Az egyes komponensek szerepe, felelőssége, hatásköre

A rendszeremet alapvetően web portál alapon tervezem létrehozni, így ez alapján próbálom meghatározni, hogy milyen komponensekre is lesz szükségem. Két részre osztom a komponenseket (kliens illetve szerver oldaliakra), ám mivel web alapú a rendszer, ezek nem különülnek el egymástól. Később ez a lista még változhat.

1.4.1. Kliens oldalról tekintve

1.4.1.1. Kapcsolat felépítés

Ez a komponens fog felelni a kapcsolat létrejöttéért a mobil eszköz böngészője és a szerver között a kliens oldalról. A mobileszköz elküldi paramétereit (elsősorban a képernyő felbontása, amely szükséges ahhoz, hogy meghatározza a szerver majd a későbbi video streaming szerver számára a küldött videó méretét).

1.4.1.2. Megjelenítés

Ez a komponens felel a felhasználói felület megjelenítéséért. A rendszer a specifikáció szerint a felhasználó mobil eszközének web böngészőjén keresztül jelenik meg, a már elküldött adatoknak (főként a felbontásnak) megfelelően optimalizálva.

1.4.1.3. Felhasználó kezelő

Ez a komponens felel a felhasználók kezelésére (be/ki jelentkezés, milyen jogai vannak a felhasználónak, stb.).

1.4.1.4. Interakció kezelő

Ez a komponens felel a felhasználók cselekvésének kezeléséért. Vagyis összefogja a különböző interakciókat.

1.4.1.5. Videó megjelenítő

Ez a komponens felel a felhasználó által lekért videó típusú (fájlformátumok még később konkretizálódnak) fájl megjelenítéséért,

lejátszásáért. Ez a komponens (valószínűleg) egy már megírt, böngészőbe beépülő multimédia lejátszó lesz. Ez még később pontosításra/tisztázásra kerül.

1.4.1.6. Feltöltő

Ez a komponens felel a felhasználó szándéka szerint feltöltendő, a mobil eszközön tárolt multimédiás tartalom (fájl) feltöltéséért a szerverre.

1.4.2. Szerver oldalról tekintve

1.4.2.1. Kapcsolat felépítés

Ez a komponens fog felelni a kapcsolat létrejöttéért a mobil eszköz és a szerver között a szerver oldalról. A szerver veszi a mobil eszköz által küldött paramétereket, beállítja a képernyő felbontást, csatlakozik a multimédiás fájlokat tartalmazó adatbázishoz, stb.

1.4.2.2. Megjelenítés

Ez a komponens felel a felhasználói felület megjelenítéséért a meghatározott felbontásnak megfelelően.

1.4.2.3. Felhasználó kezelő

Ez a komponens végzi a felhasználók kezelését a szerver oldalon (session-ök, stb.).

1.4.2.4. Interakció kezelő

Ez a komponens végzi a (böngésző által küldött) kívánt cselekvés elvégzését a szerver oldalon (például tetszési index módosítása, komment hozzáadása, stb.).

1.4.2.5. Videó stream

Ez a komponens végzi a videó elküldését a mobil eszközre.

1.4.2.6. Feltöltés kezelő

Ez a komponens végzi a mobil eszköz (felhasználó) által küldött multimédiás tartalom adatbázisba elhelyezését, illetve az (ehhez kapcsolódó) adatbázison szükséges változtatások (kategória, címkék, tetszési index, stb.) elvégzését.

1.5. A komponensek minimális információigényei, be- és kimeneti mezői és állapotváltozói

Ahogy az önálló labor weboldalán (<https://onlab.db.bme.hu/>) is olvasható: „a változóktól való függés, információ igény általában a komponensek felelősségéből, szerepköréből általában származtatható, ezért azokkal összevontan kezeljük”, ehhez alkalmazkodom is, valamint a feladat specifikációjából adódóan a komponenseket páronként (kliens- illetve szerver oldali komponensek) összevontan írom le. Ezek egyelőre elgondolások, a későbbi tervezés során még módosulhatnak.

1.5.1. Kapcsolat felépítés

A mobil eszköz elküldi a képernyőjének a felbontását, hogy optimalizálódjon a web portál valamint a multimédiás tartalmak megjelenítése. Az adatbázishoz való csatlakozás a szerver oldalon itt van eltárolva, így – mivel a megjelenítés (is) használja ezt a komponenst – már a nyitó oldal megjelenítésekor tudunk csatlakozni ahhoz.

1.5.2. Megjelenítés

Bemenete a képernyő felbontása (amit a kapcsolat felépítéséből kap paraméterként), és ez alapján jeleníti meg az oldalakat.

1.5.3. Felhasználó kezelő

Bemenetként kapja a felhasználó által a mobil eszközön begépelte felhasználónevet illetve jelszót (új session-t indít), valamint regisztráció esetén a felhasználó adatait tárolja el az adatbázisban.

1.5.4. Interakció kezelő

A felhasználó szándékától függ, hogy mik a paraméterei.

1.5.4.1. Multimédiás tartalom böngészése

A bemenetként kapott ID-jű és típusú fájlt nyitja meg (a megjelenítő komponens segítségével).

1.5.4.2. Multimédiás tartalom kedvencek jelölése

A bemenetként kapott ID-jű és típusú fájlt hozzáadja a saját (a felhasználó ID-je is bemenet) kedvencei közé, vagyis beíródik az adatbázisba.

1.5.4.3. Multimédiás tartalomhoz kommentár fűzése

A bemenetként kapott ID-jű és típusú fájlhoz kommentárt ad hozzá, vagyis a kommentár szövege beíródik az adatbázisba (természetesen a fájl és a felhasználó – bemenetként – kapott azonosítójával valamint a dátummal együtt).

1.5.4.4. Multimédiás tartalomhoz tetszési index hozzáadása

A bemenetként kapott ID-jű és típusú fájlhoz tetszési indexet (1..10) ad hozzá, ez beíródik az adatbázisba (ezek átlaga adja a fájl népszerűségét). Egy felhasználó egy fájlt csak egyszer értékelhet, ez az adatbázisba írás előtt ellenőrzésre kerül.

1.5.4.5. Felhasználó galériáinak rendezése

A felhasználó saját galériákat (könyvtárakat) hozhat létre, törölhet, átnevezhet, illetve ezekbe a galériákba helyezheti el a feltöltött fájljait. Bemenetként kapja a felhasználó azonosítóját. Minden egyes interakció után a művelet az adatbázisba íródik.

1.5.4.6. Felhasználó kedvenceinek rendezése

A felhasználó a kedvenceket is hasonlóan rendezheti, mint a galériákat (persze itt fájlok helyett csak linkek vannak), de ezeket itt törölheti/módosíthatja is. Bemenetként kapja a felhasználó azonosítóját. Minden egyes interakció után a művelet az adatbázisba íródik.

1.5.4.7. Felhasználó fájljainak kezelése

A felhasználó itt tudja módosítani (módosítás alatt az átnevezést, címkék kezelését értve), törölni az általa feltöltött fájlokat. Bemenetként kapja a felhasználó azonosítóját. Minden egyes interakció után a művelet az adatbázisba íródik.

1.5.5. Videó megjelenítő

Bemenetként kapja a képernyő felbontását, és ennek megfelelően szolgáltatja a videót a mobil eszközre, a böngészőbe beépülő video megjelenítő segítségével (ez csak egy tervezet, a tervezés során még változhat).

1.5.6. Feltöltés kezelő

Bemenetként kapja a felhasználó azonosítóját, a feltölteni szándékozott multimédiás tartalmat (fájlt), valamint annak paramétereit (név, címkék, dátum), valamint egy jelzést, hogy valamelyik nyereményjátékra is jelentkezik-e ezzel a fájlal a felhasználó.

1.6. Az egyes komponensek által használt erőforrások listája

A komponenseket itt is az előzőekhez hasonlóan összevonva kezelem.

1.6.1. Kapcsolat felépítés

A mobil eszköz és a szerver közti kommunikációs csatornát (legyen az akár 3G akár Wi-fi kapcsolat) használja adattovábbításra.

1.6.2. Megjelenítés

A mobil eszköz képernyőjén jelennek meg a kívánt tartalmak.

1.6.3. Felhasználó kezelő

Az adatbázis tartalmaz egyrészt egy „felhasználók” táblát, másrészt egy session táblát, ezek szükségesek a felhasználók kezeléséhez.

1.6.4. Interakció kezelő

A hozzászólások illetve a tetszési indexek tárolására az adatbázis tartalmaz egy „kommentárok” illetve egy „indexek” táblát. A felhasználói galériák adatainak tárolása a „galériák” táblában történik. Az, hogy a fájl mely galériában van (alapeset a felhasználó „gyökérfiókjára”), az a „fájl-a-galériában” táblában tárolódik. A kedvencek hasonlóan a galériákhoz történik.

1.6.5. Videó megjelenítő

A mobil eszköz beépített lejátszóját használja.

1.6.6. Feltöltés kezelő

A multimédiás tartalom tárolására az adatbázis tartalmaz egy „videók” táblát a video típusú fájlokra valamint egy „kép” táblát a kép típusúaknak.

1.7. A megvalósítás tervezett programnyelvi környezete és operációs rendszere

A környezet a feladat specifikációjából adódik. A tervezett rendszernek mobil eszközökön kell futnia, ezek kiszolgálására az Oracle Application Server Portal-t fogom választani, ugyanis az OracleAS Portal „kiválóan” alkalmas mobil eszközök kezelésére. Az adatbázis, amely tárolja mind a multimédiás fájlokat, mind a rendszer többi szükséges tábláját szintén Oracle rendszerben fogom tárolni. A multimédiás

fájlokhoz kihasználom az Oracle MultiMedia (vagyis korábbi nevén InterMedia) lehetőségeit. A video streaminghez pedig valószínűleg Windows Media Services-t fogok használni. A portálhoz előre láthatólag PL/SQL-t illetve vagy JSP-t (JDeveloper fejlesztővel), vagy pedig php-t fogok használni. A mobil eszközökön futó operációs rendszerek nagyobb része manapság (legalábbis az én célközönségemnél) Symbian (Nokia mobiltelefonok) vagy Windows (PDA-k). De mindegyik rendelkezik beépített web böngészővel, és ezek támogatják a Java-t, tehát probléma ebből kevés adódhat remélhetőleg, illetve amennyiben a php-nál maradok, akkor a php a szerver oldalon fut, a mobil eszközök „csak” a generált kódot kapják meg, amit a web böngészőjük szintén tud kezelni. Videó lejátszáshoz szükség van a mobil eszköznek videó lejátszóra, de ez mindegyikben van, amelyikben beépített kamera is.

2. TESZTELÉSI TERV

A tesztelési terv a rendszer állapotáról ad majd egy képet, hogy mi az, ami úgy működik, és mi az, ami funkcionalitását tekintve nem úgy, ahogy az elvárható lenne. Jó esetben választ kapunk arra, hogy mennyire áll ellen a rendszer a különböző terheléseknek, illetve mennyire használható a rendszer a (célként szánt) valós környezetében.

A tesztelést a kliens oldalról tekintem, hiszen itt történik a rendszer használata, tehát nem választom szét kliens illetve szerver oldali tesztre.

Idézve a Hallgatói önálló laboros oldalról (ami a következő címen érhető el: <https://onlab.db.bme.hu/Gondolatok/Rendszerterv>): „A tesztelési tervnek egy jórészt nem a fejlesztőknek, hanem a megrendelőnek, vagy a megrendelővel közösen kell/érdemes kialakítani.” Ez esetben (mint már korábban említettem) a rendszer valószínűleg nem fog a nagyközönség elé kikerülni, a megrendelő pedig tulajdonképpen én saját magam vagyok, tehát igyekszem így kialakítani a tesztelési tervet.

2.1. A funkcionalitás tesztelése

A funkcionalitás tesztje a legelső lépés a teszt során. E teszt azt a célt szolgálja, hogy kipróbáljuk a rendszer minden egyes komponensét, hogy a terveknek megfelelően működik-e.

Elsőként a kapcsolat felépítése, ami ugyebár a mobil eszköz web böngészőjén keresztül történik. Ha a kijelzőn megjelenik a portál, akkor a kapcsolatfelépítő illetve a megjelenítés modul a funkcióját ellátja.

Következő lépés a Regisztráció, mivel a bejelentkezéshez (illetve az oldal használatához) regisztrálni kell. A hibatűrési tesztnél kifejtem részletesebben, hogy milyen tesztvektorokkal érdemes a tesztelést elvégezni.

A portálra való belépést (bejelentkezést) követően a tesztelést két részre bonthatjuk: a böngészésre és a feltöltésre.

Feltöltés tesztjeként a mobil eszközön tárolt különböző típusú, méretű multimédiás fájlokat töltjük fel a rendszerre. Ezekről részletesebben szintén a következő pontokban írok.

Böngészés tesztjénél különböző tartalmakat jelenítünk meg, a teszt célja, hogy megfelelően jelennek-e meg a videók (megfelelően elindul a mobil eszköz beépített

lejátszója, és lejátszódik-e a fájl) illetve a képek megjelennek-e? Ide tartoznak még a különböző kommentárok hozzáadása, osztályozások, stb.

Érdeemes még a moderátorok szerepét tesztelni. Tudnak-e törölni „nem odaillő tartalmakat” a rendszerből, illetve a kommentárokat tudják-e megfelelően módosítani, esetlegesen felhasználókat figyelmeztetni, kitiltani.

2.2. Hibatűrési tesztek

A hibatűrési teszt több részből áll(hat). Tesztelni kell egyrészt, hogy a nem rendeltetésszerű használatra miképp reagál a rendszerünk (legyen az egy képzetlen felhasználó, vagy akár egy hirtelen fellépő nagymértékű terhelés). Ennek a tesztnek része még a rosszindulatú támadások elleni teszt, hogy mennyire is „ellenségtűrő” a rendszer.

A kapcsolat felépítésével nem foglalkozom külön, hiszen ha egy felhasználó a mobil eszközét használva eljut a portálig, akkor vélhetően már tudja, hogyan kell kezelni az eszköze web böngészőjét.

A regisztrációkor hibatűrési teszt tesztvektoraként érdemes megadni különböző SQL utasításokat a felhasználók létrehozására. Vagyis, például, hogy az email címük megfelelő formátumú-e ([x@y.z](#)), minden adatot kitöltött-e, az adatokban vannak-e speciális karakterek (miket fogadunk el és miket nem).

Feltöltés tesztjénél különböző típusú és méretű fájlokat töltünk fel. Tesztelni kell, hogy a rendszer miképpen reagál, ha nem megfelelő típusú (a fájl kiterjesztése megtévesztő lehet) fájlt töltünk fel. Tesztelnünk kell azt is, hogy mekkora méretű fájlokat tudunk feltölteni, vagyis mi történik, ha át akarjuk lépni a maximálisan megengedett feltölthető méretet? Sajnos a tartalom automatikus kiszűrése („nem odaillő tartalom”) nem áll módunkban, így ez nem része a tesztelésnek.

Ugyancsak a feltöltés teszteléséhez tartozik, hogy megfelelően beállíthatóak-e a feltöltött tartalom adatai (név, címkék, stb.). Hibatűrésnél érdemes kipróbálni (tesztvektorként), hogy milyen speciális karaktereket fogad el a rendszer, mekkora maximális karaktersorozat tud kezelni, illetve mi történik akkor, ha két fájlnak ugyanazt a nevet adjuk.

A böngészés hibatűrési tesztjénél a kommentárok illetve az osztályozást tekinthetjük, hogy megfelelően adja hozzá a rendszer a kommentárt, illetve számolja az átlagokat.

A moderátorok szerepének tesztelésénél hibatúrési szempontból tesztelni kell, hogy mi történik, ha egy moderátor olyan tartalmat akar törölni (tesztvektorként szolgáló SQL utasítással megadva), ami már nem is létezik a rendszerben. Hasonlóan így a kommentárookra, felhasználókra vonatkozólag.

2.3. Kiszolgálási idő tesztje

A kiszolgálási idő nagyban függ attól, hogy honnan (földrajzi szempontból) is használjuk a rendszert, vagyis milyen sebességű internet kapcsolaton keresztül. Egy mobiltelefon esetében például hol van, és milyen a 3G lefedettség, vagy éppen egy Wifi-t használó PDA milyen sebességgel tud csatlakozni a vezeték nélküli hálózatra.

A kiszolgálási idő elsősorban a nagyobb méretű multimédiás tartalmak fel illetve letöltésénél játszik főszerepet, hiszen ha nem elég a sávszélesség, akkor a videó szaggatottá válik, a képre pedig túlon túl sokat kell várnunk.

Ezen okoknál fogva a kiszolgálási idő tesztje különböző helyeken, különböző kapcsolatú eszközökkel való fel illetve letöltés lesz.

2.4. Valós környezeti teszt

A valós környezeti teszt tulajdonképpen a kiszolgálási idő tesztjéhez hasonlóan (esetleg vele egybevéve) fog történni.

3. ÜZEMELTETÉSI TERV

„Az üzemeltetési terv készítésének kiindulási pontja az a helyzet, amikor a rendszer – előzetesen, fejben és papíron – működik.” (idézet a Hallgatói önálló laboros oldalról, <https://onlab.db.bme.hu/Gondolatok/Rendszerterv>)

3.1. A rendszert felügyelő felhasználók „howto”-ja

A rendszernek (alapfunkcióiból kiindulva) szüksége van egy úgynevezett adminisztrációs felületre (interfészre), a külső (adminisztrátori, moderátori) beavatkozási lehetőségekhez.

3.1.1. Monitorozás

A rendszer része lesz egy naplófájl, amely az adatbázisban tárolódik. A rendszer minden eseményről bejegyzést tesz a naplóba. Ilyen események a különböző felhasználókat érintő főbb események (regisztráció, belépés, kilépés, regisztráció törlése, stb.), a felhasználók interakciói (kommentár hozzáadása, osztályozás, stb.), a feltöltött tartalmakkal kapcsolatos események (feltöltés, törlés, módosítás, stb.) valamint statisztikai adatokat (regisztráltak száma, látogatók száma, videó megjelenítések száma, stb.) is bizonyos időközönként.

3.1.2. Karbantartás

Karbantartás során a rendszer moderátorai az adminisztrációs felület rájuk vonatkozó részén megtekintik a napló bejegyzéseit, és ezek alapján tudják elvégezni feladataikat. Ilyen például a nem megfelelő tartalmú multimédiás tartalmak törlése illetve a kommentárok ellenőrzése (trágárság, stb).

Az rendszer adminisztrátorok tevékenysége tulajdonképpen csak a rendszer működőképességének megőrzésére terjed ki, mivel az egyéb teendőket a moderátorok végzik. Vagyis az adminisztrátorok végzik többek között a biztonsági mentéseket (szükség esetén visszaállításokat), valamint az ő feladatuk a rendszer háttéréül szolgáló környezet üzemeltetése.

3.2. A rendszer felhasználását segítő tréningek, tanfolyamok

Mivel a rendszer nagy valószínűséggel nem fog kikerülni a hétköznapi életbe, és csak az Önálló laboratórium keretei között fog működni, a különböző tréningek,

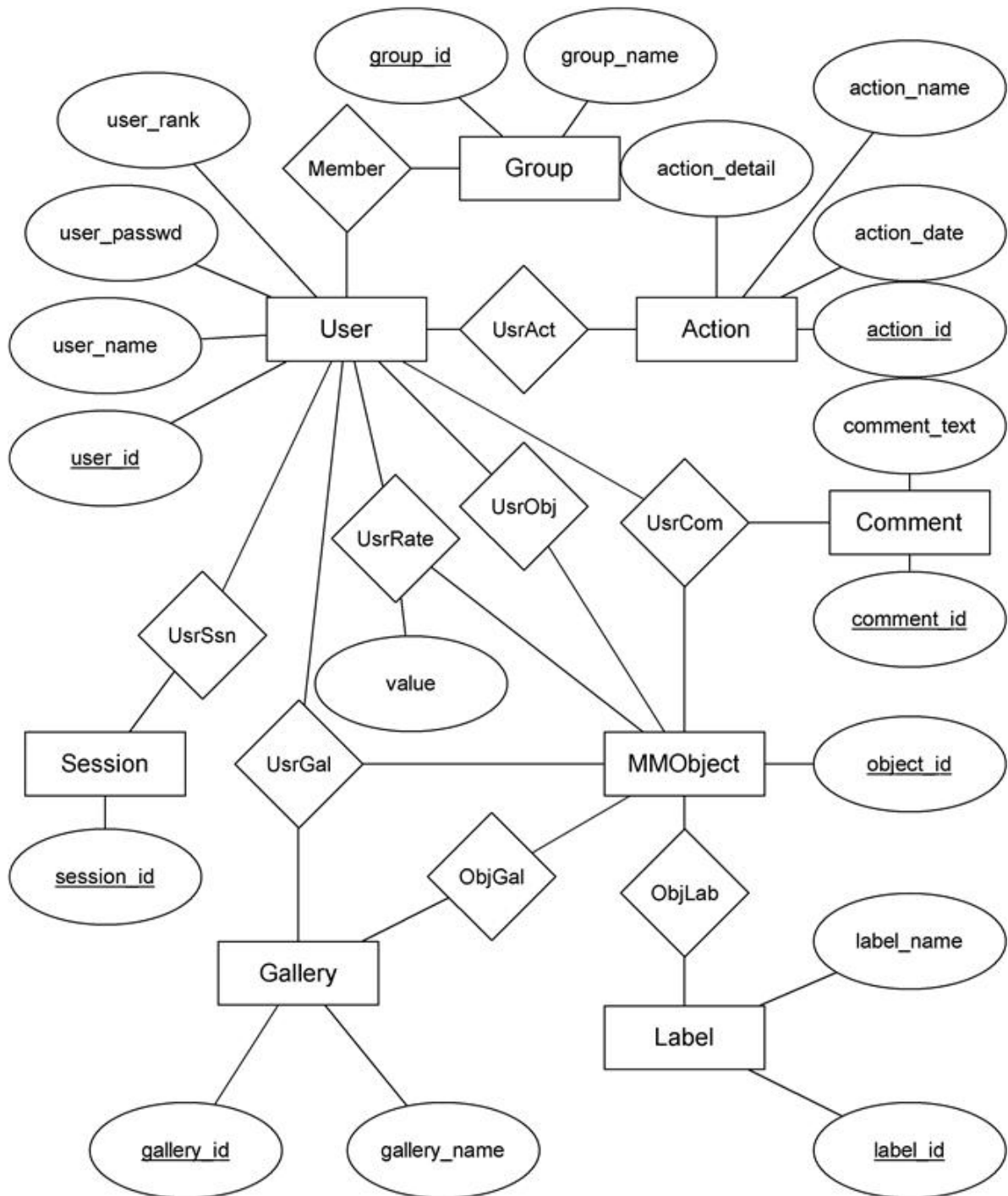
tanfolyamok programjának kialakítása illetve segédanyagok elkészítése nem fog most megtörténni. Természetesen, amennyiben mégis a nagyközönség elé fog kerülni a rendszer, akkor ezek a feladatok is el lesznek végezve.

4. ADATSZERKEZETI TERV

4.1. Az adatbázis terv

Az adatbázis terv a passzív erőforrások egységes leírását foglalja magába. Nem csak az adatbázis, hanem a „naplófájl” definícióját is tartalmazza. A napló az adatbázis része lesz (Action tábla), tehát ezt nem részletezem külön.

4.1.1. Entitás-relációs diagram



Elsőként az Entitás-relációs diagramot készítettem el az adatbázis megtervezése során. Sajnos az általam használt OpenSource diagramkészítő (UML Diagrammer) nem támogatja teljesen jól az ER diagramok készítését, ezért a diagram nem teljesen igazodik a konvenciókhoz.

Az MMLObject entitás a Multimédiás Objektum, melynek attribútumait nem részleteztem az ábrán (ez az Oracle MultiMedia sajátossága). Ebből fog származni a kép típus, melynek attribútumai: méret, felbontás (bit/pixel), tömörítési típus, tartalom formátuma. Ugyancsak ebből fog származni a videó típus, melynek attribútumai: név, leírás, fájl formátum, MIME típus, enkódolás típusa audio csatornák száma, minta/keret arány, keret felbontása és a keretek száma, bit ráta, tömörítés típusa, hossz.

4.1.2. Séma

Az ER diagramból készítettem el az adatbázis sémáját.

- User(user_id, user_name, user_passwd, user_rank)
- Group(group_id, group_name)
- Member(user_id, group_id)
- Action(action_id, action_name, action_date, action_detail)
- UsrAct(user_id, action_id)
- Session(session_id)
- UsrSsn(user_id, session_id)
- MMLObject(object_id)
- UsrObj(user_id, object_id)
- UsrRate(user_id, object_id, value)
- Comment(comment_id, comment_text)
- UsrCom(user_id, object_id, comment_id)
- Label(label_id, label_name)
- ObjLabel(object_id, label_id)
- Gallery(gallery_id, gallery_name)
- UsrGal(user_id, gallery_id)
- ObjGal(object_id, gallery_id)

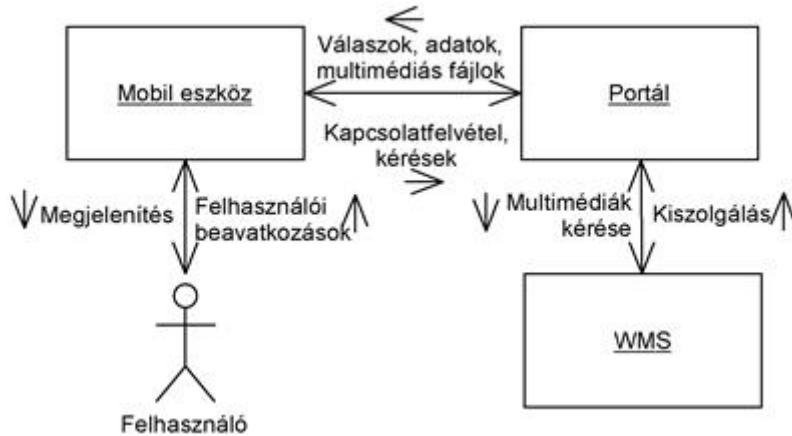
5. MEGVALÓSÍTÁSI TERV

5.1. Osztálydiagramok

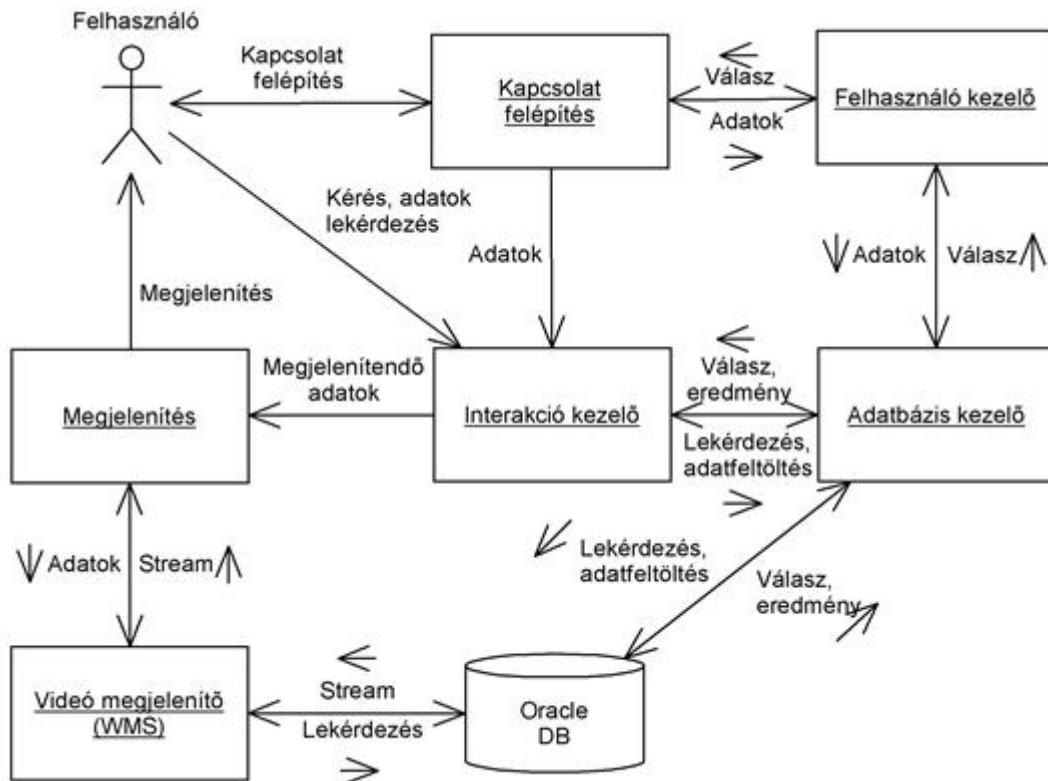
Az osztálydiagramok elkészítését a jövő félévre csúsztatom át. Az implementáció során várhatóan többször változni fognak az osztálydiagramok, a minél hatékonyabb megvalósítás érdekében.

5.2. Blokkvázlatok

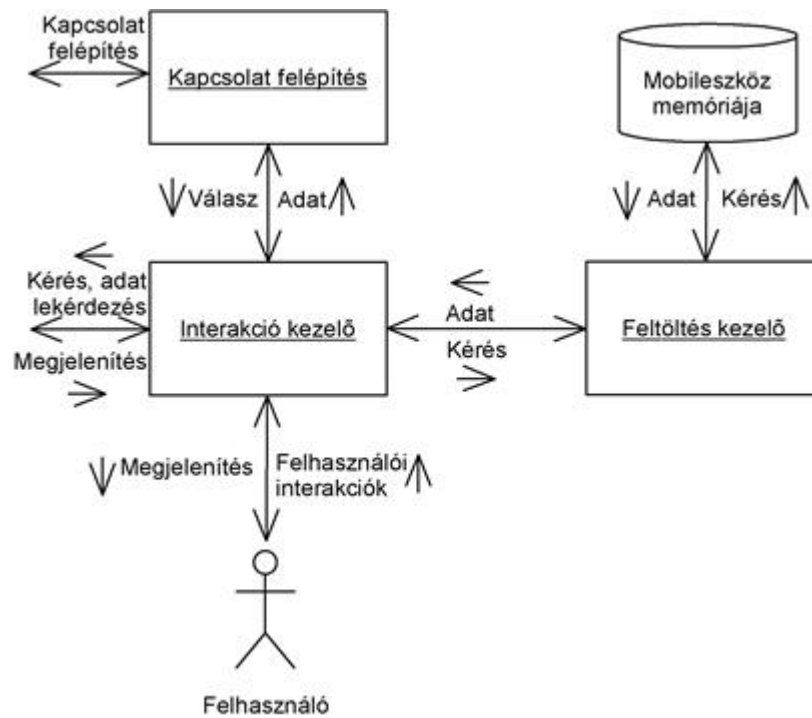
5.2.1. A teljes rendszer blokkvázlata



5.2.2. A szerver oldal blokkvázlata



5.2.3. A kliens oldal blokkvázlata

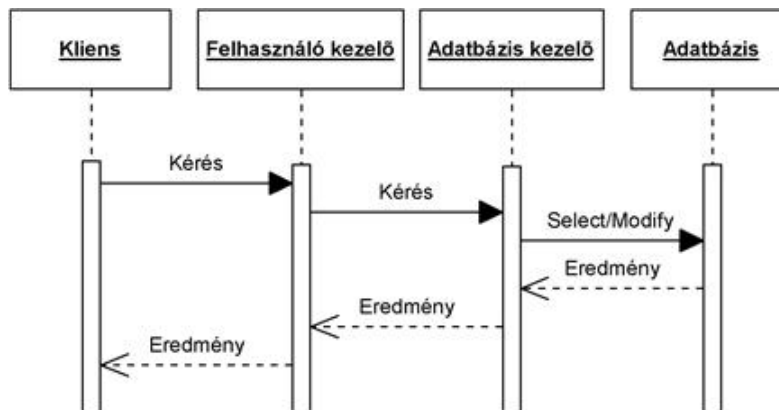


5.3. Szerver oldali diagramok

5.3.1. Szekvencia diagram

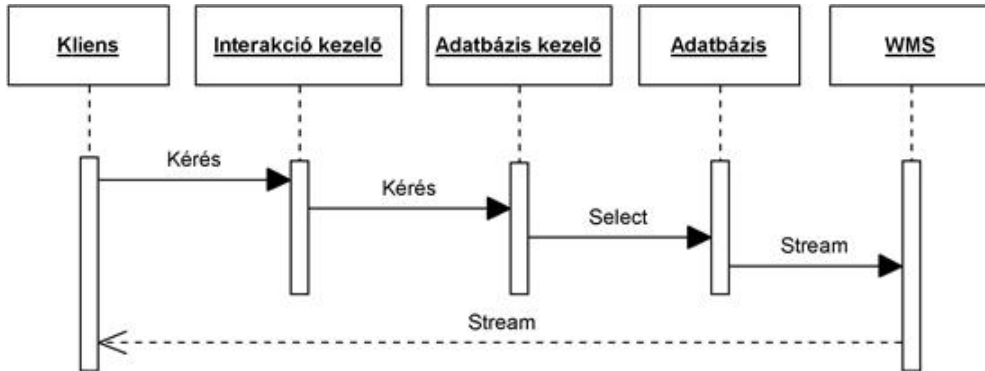
5.3.1.1. Belépés/Regisztráció

Felhasználó belépése, regisztrációja.



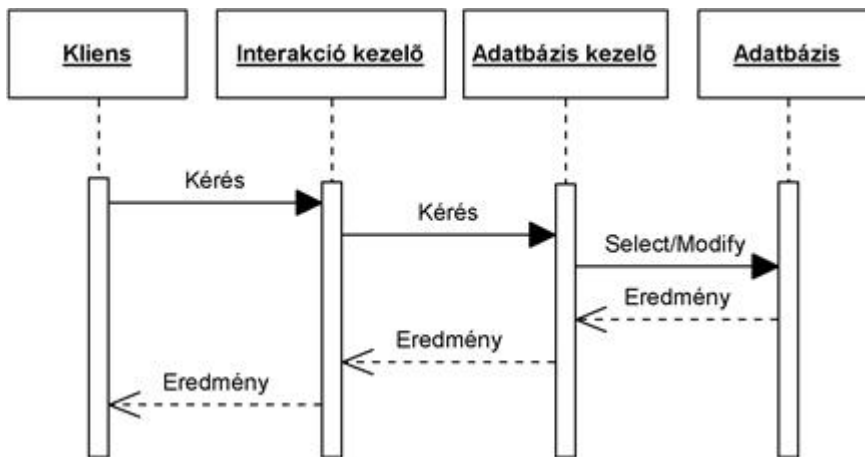
5.3.1.2. Video streaming

Videó streaming a mobil eszközre.



5.3.1.3. Interakciók

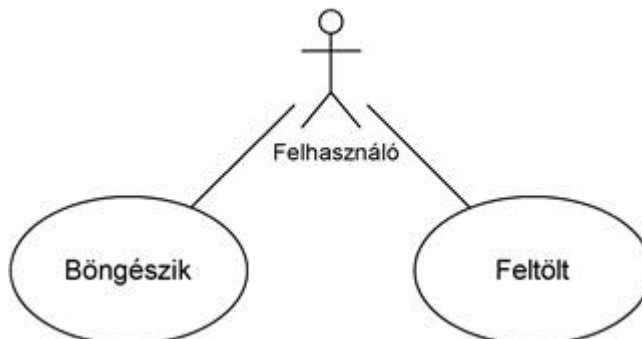
Minden egyéb felhasználói interakció.



5.4. Kliens oldali diagramok

5.4.1. Use Case diagram

A kliens (egy felhasználó) egyszerűen egy Use Case diagrammal jellemezhető.



5.4.2. State machine

A kliens (egy felhasználó) részletesebben oldal legjobban egy állapotgéppel (state machine) jellemezhető. A csatlakozás (indítás) után a felhasználó, ha

még nem regisztrált, akkor regisztrál, egyébként belép. Belépés után eldöntheti, hogy épp böngészni fog, vagy feltölt. Ha böngészik, miután megnézte az adott tartalmat, eldöntheti, hogy kíván-e kommentárt fűzni, avagy osztályozni, vagy egyszerűen továbblép. Ezután illetve a feltöltés után vagy kilép, vagy kezdi előlről.

