

Webes alkalmazások fejlesztése - 1. beadandó

Nagy András

April 12, 2016

1 Feladat

Készítsünk egy aukciókkal foglalkozó online rendszert, ahol különböző tárgyakra licitálhatnak a felhasználók. A webes felületet a licitálók használhatják a tárgyak megtekintésére, illetve ajánlattételre.

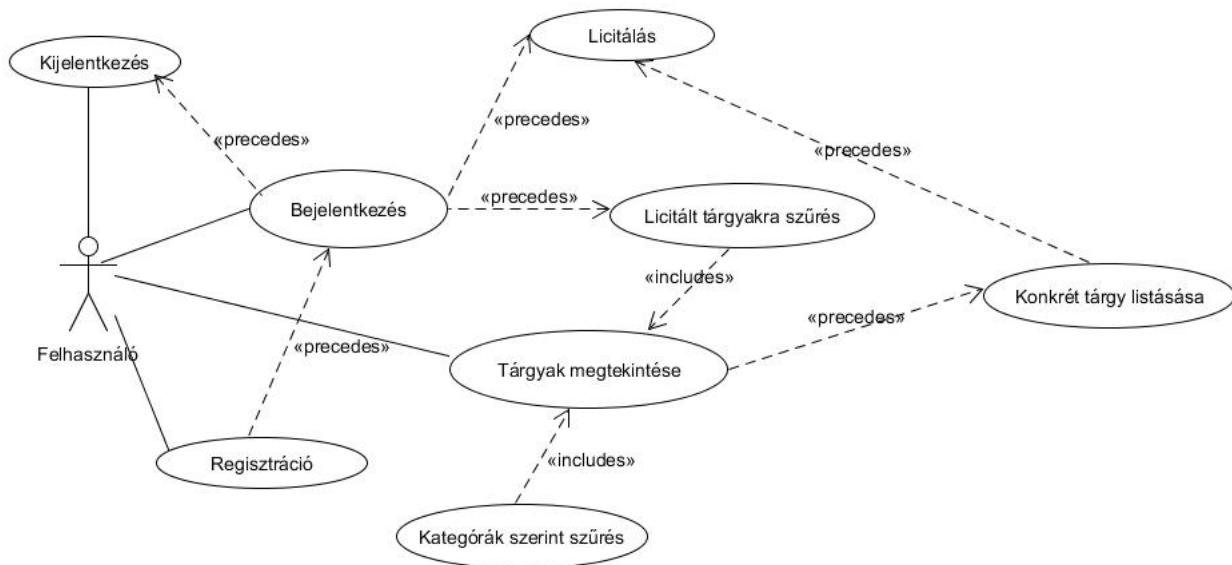
- A főoldalon a legutoljára meghirdetett 20 tárgy listázódik (név, hirdető, jelenlegi licitösszeg), de lehetőségünk van kategóriánként megtekinteni az összes (még aktív) hirdetést. Egy oldalon legfeljebb 20 tárgy látható (a meghirdetés dátuma szerint csökkenő sorrendben), az oldalak között lapozni lehet. A lista szűrhető név(részlet)-re. A tárgyat kiválasztva megjelennek a részletes adatok (kép, leírás, lezárás és meghirdetés dátuma , aktuális licit).
- A licitálónak előbb regisztrálnia kell az oldalon (név, telefonszám, e-mail cím, felhasználónév, jelszó, megerősített jelszó), majd ezt követően bejelentkezhet. A bejelentkezett felhasználó kijelentkezhet.
- Bejelentkezést követően érhető el a licitálás minden aktív tárgynál. Licitáláshoz ki kell jelölni a tárgyat és meg kell adni az összeget. Első licit esetén az összegnek a minimális licitnek kell lennie, később pedig mindenképpen nagyobbnak kell lennie a korábbi liciteknél. Egy felhasználó tetszőlegesen sokszor licitálhat egy tárgyra. A licitet visszavonni nem lehet.
- A felhasználó külön listázhatja azokat a tárgyakat, amelyekre legalább egyszer licitált. A listában külön megjelöljük az aktív tárgyakat, valamint azokat, ahol vezeti a licitet

2 Feladat elemzése

A feladatot a három rétegű, Model-View-Controller architektúrában valósítjuk meg.

- Az adatok tárolásához, és a felhasználó kezeléshez létrehozunk egy adatbázist, melyben a tárgyak, a licitek, valamint a felhasználók adatait tárolhatjuk. Az adatbázis modelljét az AuctionPortalEntities entitás modell biztosítja.
- Az oldal egységes megjelenítésért létrehozunk egy megosztott megjelenítőt, melyben be és kijelentkezhetünk, láthatjuk a tárgyakat kötegórák szerint és szűrhetünk is rájuk.
- A Home könyvtár alatt létrehozzuk a tárgyak listázásért felelős nézetet, valamint az adott tárgy részletei nézetet.
- Külön nézetet hozunk létre a licitáláshoz, a regisztrációhoz és a bejelentkezéshez. Mindezekhez létrehozunk egy-egy nézetmodellt, melyben eltárolhatjuk a felhasználó által kitöltött információkat.
- A kontrollerek felelőssége csökkentése érdekében létrehozunk két funkcionalitásért felelős osztályt, az AccountService-t, mely a felhasználói műveletekért felelős, valamint az AuctionService-t, amely a konzisztens licitálást biztosítja, valamint a megfelelő licitek és tárgyak elérést biztosítja.
- Azon funkciókat, melynek minden controllernek végre kell hajtania, mint a kategóriák meghatározása, kiemeljük egy bázis controllerbe.
- Külön controller felel a felhasználói műveletekért, valamint a licitálás kezeléséért.

3 Felhasználói estek



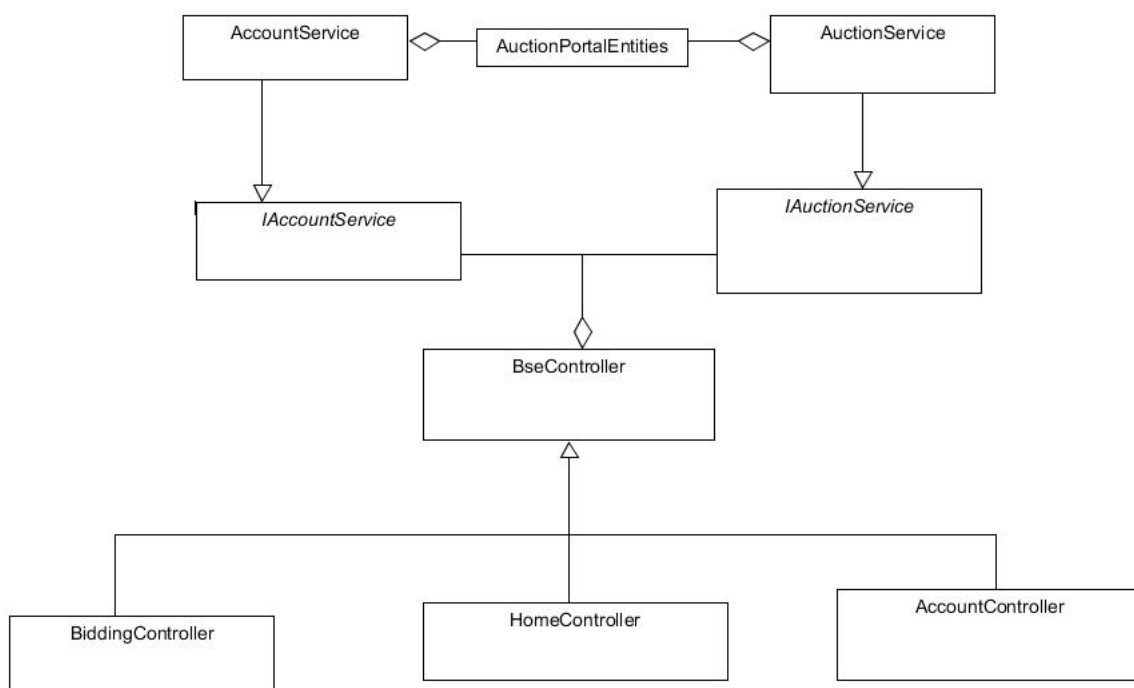
4 A rendszer szerkezete

4.1 Komponensek

Az MVC architektúrának megfelelően a programnak 3 fő komponense van.

- A Models névtér tartalmazza az adatbázis alapján generált entitásmodell, a nézetmodelleket, az felhasználókezelését, valamint az adatbázis műveletekért felelős osztálymodelleket, valamint a validációért felelős statikus osztályokat.
- A Contorllers névtéren belül a kontrollerek helyezkednek el. A program 3 különböző kontrollere oszlik, valamint a közös funkcionalitást biztosító bázis kontrollere.
- A View-en belül helyezkedik el a közös megjelenítésért felelős nézet, a tárgyak listázásáért felelős főoldali nézet, a be-ki jelenkezésért és a regisztrációért felelős nézetek, valamint a tárgyak részleteit megadó, és a licitálás felület.

4.2 Osztályok



5 Adatbázis felépítése

