

Webes alkalmazások fejlesztése - 1. beadandó

Iványi Patrik

April 19, 2018

1 Feladat

Készítsük el egy étel-futár vállalat rendeléseket kezelő rendszerét.

- A weblap főoldalán megjelennek a kategóriák (pl. levesek, pizzák, üdítők), valamint a 10 legnépszerűbb (legtöbbször rendelt) étel/ital.
- A kategóriát kiválasztva listázódnak a tételek (név és ár kíséretében), amelyek szűrhetők név(részlet)re. Ételek esetén leírás is van. Külön meg vannak jelölve a csípős, illetve vegetáriánus ételek.
- Ételek és italok tetszőleges számban helyezhetők a kosárba egy adott felső korlátig (20.000 Ft), afelett több terméket nem lehet a kosárba helyezni.
- A kosár tartalma bármikor megtekinthető, ekkor látszódnak a felvett tételek, illetve látható az összár. Bármely tétel kivethető a kosárból.
- A rendelést törölhetjük, illetve leadhatjuk. Utóbbi esetben meg kell adnunk a nevünket, címünket, illetve telefonszámunkat, majd elküldhetjük a rendelést.

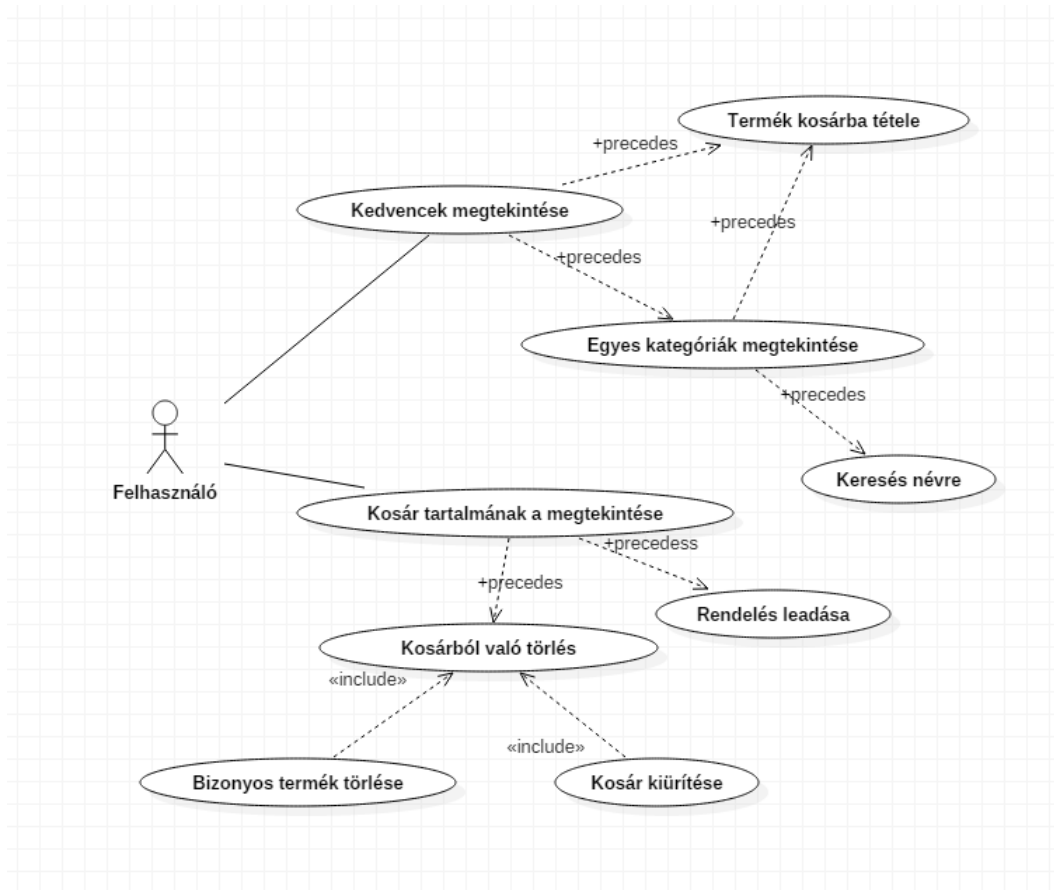
2 Feladat elemzése

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

- Az adatok tárolásához, és a felhasználó kezeléshez létrehozunk egy adatbázist, melyben az ételek/italok, a kategóriák, a rendeléseket, valamint a vásárlás során a kosár tartalmát tároljuk. Az adatbázis modelljét az RestaurantContext biztosítja.

- Az oldal egységes megjelenítésért létrehozunk egy megosztott megjelenítőt.
- A Home könyvtár alatt létrehozuk a kedvenc ételek/italok listázásért felelős nézetet, valamint az oldalsó menüsávban egy lehetőséget, hogy az adott felhasználó szűrhesse kategóriára.
- Külön nézetet hozunk létre a kategóriák megjelenítése érdekében, a nézet a kiválasztott kategória alapján változtatja tartalmát.
- 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.
- A bevásárló kosarat egy ShoppingCart modell segítségével valósítottuk meg, mely az adott session és az adatbázis között épít ki kapcsolatot.
- Külön controller felel a vásárlás lebonyolításáért, valamint az ételek közti keresésekért.

3 Felhasználói estek

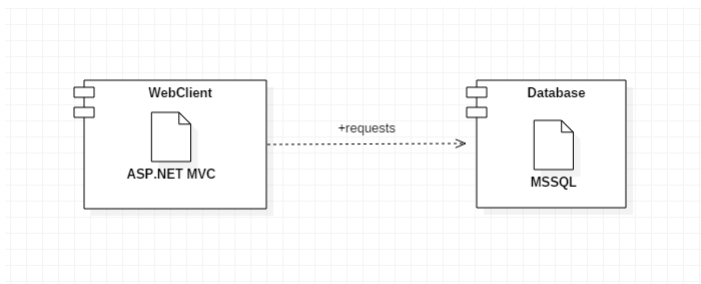


4 A rendszer szerkezete

4.1 Komponensek

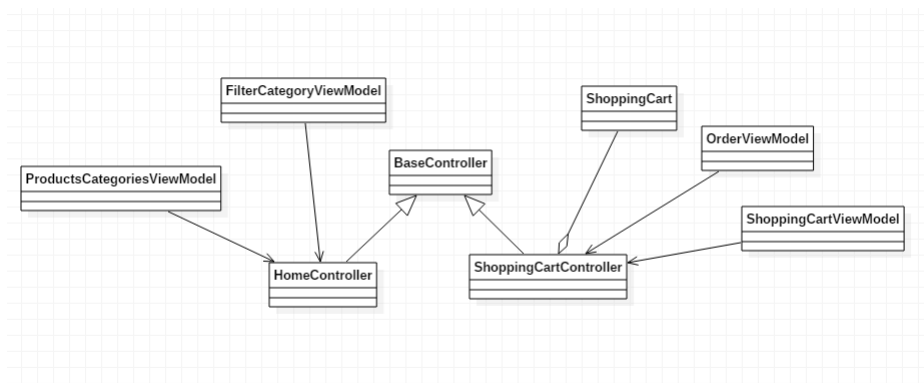
Az MVC architektúrának megfelelően a programnak 2 fontos komponense van.

- Az adatbázis felelős a rendszer által használt adatok, és a bevérlés lebonyolításához történő információk tárolására. Az adatbázishoz való kapcsolódást az Entity Framework segítségével biztosítjuk.
- A webes alkalmazási felület ASP.NET-ben MVC architektúrával megvalósítva, amely tartalmaz felhasználói nézeteket és ehhez kapcsolódó nézetmodelleket, kontrollereket, valamint az adatbázis modelljét, és a funkcionalitásért felelős kiszolgáló osztályokat.



4.2 Osztályok

- Külön project felelős az adatbázis modell megvalósításáért, mely kapcsolatban van az alkalmazást megvalósító modellel.
- A Models névtéren belül helyezkednek el a nézetmodelleket, valamint a bevásárlást végző osztályok.
- A Controllers névtéren belül a kontrollerek helyezkednek el. A program 2 különböző kontrollere oszlik, valamint a közös funkcionalitást biztosító bázis kontrollere.
- A View-en belül helyezkedik el a közös megjelenítésért felelős nézet, a kedvencek listázásáért felelős főoldali nézet, a kosár tartalmát megjelenítő és a rendelés leadásáért felelős nézetek, valamint az egyes kategóriákhoz tartozó ételek/italokrészleteit megadó felület



5 Adatbázis felépítése

- Az adatbázisban tároljuk az egyes ételeket/italokat.
- A különböző kategóriákat

- Az adott felhasználó által leadott rendeléseket
- A bevásárlás lebonyolításához szükséges adatokat

