Pannon Egyetem   
Műszaki Informatikai Kar  
Rendszer- és Számítástudományi tanszék  
mérnökinformatikus BSc

SZAKDOLGOZAT

dolgozat címe

Rozsenich Balázs

Témavezető: Frits Márton

2015.

Feladatkiírás:

Nyilatkozat

Alulírott *Rozsenich Balázs* hallgató, kijelentem, hogy a dolgozatot a Pannon Egyetem *Rendszer- és Számítástudományi tanszék* tanszékén készítettem a *mérnökinformatikus* végzettség megszerzése érdekében.

Kijelentem, hogy a dolgozatban lévő érdemi rész saját munkám eredménye, az érdemi részen kívül csak a hivatkozott forrásokat (szakirodalom, eszközök, stb.) használtam fel.

Tudomásul veszem, hogy a dolgozatban foglalt eredményeket a Pannon Egyetem, valamint a feladatot kiíró szervezeti egység saját céljaira szabadon felhasználhatja.

dátum (Veszprém, 2008. február 31. formátummal)

aláírás

Alulírott *Frits Márton* témavezető kijelentem, hogy a dolgozatot *Rozsenich Balázs* a Pannon Egyetem *Rendszer- és Számítástudományi tanszék* tanszékén készítette *mérnökinformatikus* végzettség megszerzése érdekében.

Kijelentem, hogy a dolgozat védésre bocsátását engedélyezem.

dátum (Veszprém, 2008. február 31. formátummal)

aláírás

Köszönetnyilvánítás

Tartalmi összefoglaló

Az összefoglalónak tartalmaznia kell (rövid, velős és összefüggő megfogalmazásban) a következőket:

• téma megnevezése,

• megoldott feladat megfogalmazása,

• megoldási mód,

• elért eredmények,

• **Kulcsszavak:** (4-6 darab)

Abstract

Tartalom

[Feladatkiírás: 2](#_Toc416185777)

[Nyilatkozat 2](#_Toc416185778)

[Köszönetnyilvánítás 4](#_Toc416185779)

[Tartalmi összefoglaló 5](#_Toc416185780)

[Abstract 6](#_Toc416185781)

[Tartalom 7](#_Toc416185782)

[1 Bevezetés 10](#_Toc416185783)

[1.1 A probléma és megoldása 10](#_Toc416185784)

[2 Szálláskereső portálok 11](#_Toc416185785)

[2.1 Szallas.hu 12](#_Toc416185786)

[2.2 Booking.com 12](#_Toc416185787)

[2.3 Trivago.hu 12](#_Toc416185788)

[2.4 Konklúzió 13](#_Toc416185789)

[3 Nemlineáris programozás 13](#_Toc416185790)

[4 Ruby on Rails 13](#_Toc416185791)

[5 Specifikáció 13](#_Toc416185792)

[5.1 Funkcionális követelmények 14](#_Toc416185793)

[5.1.1 Szereplők 14](#_Toc416185794)

[5.1.2 Szobák szűrése 14](#_Toc416185795)

[5.1.3 Szobafoglalás 14](#_Toc416185796)

[5.1.4 Értékelés 15](#_Toc416185797)

[5.1.5 Intelligens keresés 15](#_Toc416185798)

[5.1.6 Törzsadatok 15](#_Toc416185799)

[5.1.7 Tartós címek 15](#_Toc416185800)

[5.2 Célcsoport 15](#_Toc416185801)

[6 Tervezés 15](#_Toc416185802)

[6.1 A rendszerben megjelenő fő folyamatok 16](#_Toc416185803)

[6.1.1 Szobafoglalás 16](#_Toc416185804)

[6.1.2 Foglalás visszaigazolás 17](#_Toc416185805)

[6.1.3 Intelligens keresés 18](#_Toc416185806)

[6.2 Nemlineáris programozási modellek 18](#_Toc416185807)

[6.2.1 Olcsó szobák 18](#_Toc416185808)

[6.2.2 Közeli szobák 18](#_Toc416185809)

[6.2.3 Olcsó és közeli szobák 18](#_Toc416185810)

[6.3 Adatbázis tervezet 19](#_Toc416185811)

[6.4 Technológia 19](#_Toc416185812)

[6.4.1 Ruby on Rails 19](#_Toc416185813)

[6.4.2 PostgreSQL 19](#_Toc416185814)

[6.4.3 AMPL 19](#_Toc416185815)

[6.4.4 Bonmin 19](#_Toc416185816)

[6.4.5 HTML, CSS, Javascript 19](#_Toc416185817)

[6.4.6 Gems… 19](#_Toc416185818)

[7 Megvalósítás 19](#_Toc416185819)

[7.1 Autentikáció és autorizáció 19](#_Toc416185820)

[7.2 Szobák szűrése 19](#_Toc416185821)

[7.3 Intelligens keresés 20](#_Toc416185822)

[7.4 Szobafoglalás 20](#_Toc416185823)

[8 Felületek és használat 20](#_Toc416185824)

[8.1 Menüsáv 20](#_Toc416185825)

[8.2 Szobák 20](#_Toc416185826)

[8.3 Szálláshelyek 20](#_Toc416185827)

[8.4 Foglalások 20](#_Toc416185828)

[8.5 Intelligens keresés 20](#_Toc416185829)

[8.6 Kosár 20](#_Toc416185830)

[8.7 Adminisztrációs felületek 20](#_Toc416185831)

[9 Tesztelés 20](#_Toc416185832)

[9.1 Tesztelési környezet 21](#_Toc416185833)

[9.2 Teszt adatok 21](#_Toc416185834)

[9.3 Teszt eredmények 21](#_Toc416185835)

[10 Összefoglalás 21](#_Toc416185836)

[Irodalomjegyzék 22](#_Toc416185837)

[Mellékletek 23](#_Toc416185838)

[CD Melléklet 24](#_Toc416185839)

# Bevezetés

A szakdolgozatom témája egy olyan webalkalmazás elkészítése, amely csoportok (pl.: osztályok, baráti vagy üzleti társaságok) számára teszi egyszerűbbé a több szálláshelyen történő szobafoglalás menetét és kezelését. Az alkalmazás szempontjából fontos az internetes platform, mert így lehet a legolcsóbban a legszélesebb felhasználói körnek elérhetővé tenni. A projekt munkacímének a *VAGATO* szót választottam, amelyet a katalán *vaganto* (jelentése: barangolás) szóból képeztem.

## A probléma és megoldása

A csoportos turizmus jelentős szereppel bír a turizmusban, gondoljunk csak a tavasszal és ősszel, százával kirándulni induló diákokra, a közös szórakozásra vágyó baráti társaságokra, vagy az egyéb, üzleti célból utazó társaságokra. Egy csoport számára, főleg szezonban kivételesen nehéz mind árban, mind távolságban megfelelő szálláshelyet találni, illetve gyakran előfordul, hogy egy szálláshely nem képes megfelelő számú kapacitást kínálni. A kapacitás korlátja lehet az aktuális foglaltság miatt, vagy, jellemzően kisebb településeken, a szálláshelyek alapvető szobakínálatának csekélysége. Ilyen helyzetekben az utazásszervező feladata az, hogy összegyűjtse a szálláshelyek ajánlatait és az idővel versengve kalkulációk útján kiválassza a megfelelő szálláshelyek megfelelő szobáit.

A szálláshelyek kiválasztása után az utazásszervező szembesül a következő a problémával. Minden szálláshely egyedileg kezeli a foglalásokat, az utazásszervezőnek minden szálláshellyel külön-külön kell megegyeznie. Ez rengeteg, egymástól független ügyintézést és papírmunkát jelent és jelentősen megbonyolítja a folyamatot.

Az általam tervezett webalkalmazás a fent vázolt problémákat igyekszik feloldani és használható megoldást kínálni. A koncepció az, hogy a jelenleg szálláshely orientált piacot meg kell fordítani és a középpontba a szobákat kell helyezni. A szobának, csakúgy, mint a légkondicionálás vagy az ellátás, csak egy tulajdonsága az, hogy mely szálláshelyhez tartozik. A szálláshelyek adta kötöttségek feloldásával már könnyű elképzelni egy olyan portált, ami a szobákat, mint egy *webshop*-ban, termékekként sorolja fel. A szobák a szálláshelyektől függetlenül kereshetők, szűrhetők és foglalhatók. A szobák e fajta individuális termékként való kezelése a kulcs ahhoz, hogy az utazásszervező olyan foglalásokat tudjon összeállítani, amiben egyszerre jelenik meg több szálláshely több szobája egy közös felületen.

A szobafoglalás folyamatát tehát most már el lehet képzelni úgy, hogy az utazásszervező a portált böngészve, egy virtuális kosárba helyezi a kellő szobákat. A böngészés végén a kosarában lévő szobákat egy foglalássá egyesíti és a vendégadatok megadása után véglegesíti azt.

A szobák kiválasztásának folyamata bonyolult, azonban jól automatizálható. A keresés szempontjai kitérnek a felszereltségre, a szolgáltatásokra és az elérhetőségre. Ezek a feltételek gyorsan és egyszerűen szűrhetők úgy, hogy az utazásszervező egy űrlapon megjelöli a kívánalmakat. A nagyobb nehézséget az ár, a minőség és a távolság feltételei adják. Az utazásszervező olyan szobákat akar, amik olcsók, ugyanakkor nincsenek távol egymástól; vagy a távolság nem számít, de legyenek minél jobb értékelésű szálláshelyeken. Az efféle szempontokhoz már nem elég szimplán sorrendbe állítani a szobákat és kiválasztani az első *N* darabot. Az optimális megoldás kísérletezés útján kézzel is elvégezhető, azonban kimondottan időigényes feladat. A webalkalmazásnak tehát rendelkeznie kell egy olyan funkcióval, ahol a kényelmi szempontok és a csoport létszáma szerint egy ár, távolság illetve minőség szerint optimális megoldást kap az utazásszervező arról, hogy mely szobákat kell lefoglalnia. A felvázolt funkciót a rendszerben *intelligens keresés*nek neveztem el.

# Szálláskereső portálok

Ebben a fejezetben a magyar szálláskereső piac legnépszerűbb portáljait vizsgálom meg a szerint, hogy milyen lehetőségeket kínálnak a szobák, illetve szálláshelyek keresésére, szűrésére. A vizsgálat tárgya továbbá, hogy mennyire támogatják a csoportos szálláskeresés 1.1 fejezetben bemutatott problémáit.

## Szallas.hu

A szallas.hu egy magyar alapítású és fejlesztésű szálláskereső portál, amely 2007 óta üzemel. A szallas.hu tekinthető a magyar szálláskereső piac legnépszerűbb szereplőjének. A szállásadók részére egységes megjelenést és könnyű foglalást ígér jutalékért cserébe.

A szálláskeresés során részletesen megadhatók a keresés feltételei hely, ár és szolgáltatások terén is. A találati listában szálláshelyek láthatók, egy szálláshelyet kiválasztva válnak láthatóvá az ajánlott szobák. Az utazó személyeket 30 felnőtt és 10 gyerek számosságban maximalizálták a keresés során. Egy foglalás csak egy szálláshely kínálatát tartalmazhatja.

A portál rendelkezik értékelési rendszerrel.

## Booking.com

A booking.com egy nemzetközi szálláskereső portál, amely 2011 lépett be a magyar szálláskereső piacra. A szallas.hu közvetlen riválisaként tekinthető, szolgáltatásaik megegyeznek. A szálláskeresők körében alacsonyabb népszerűséggel bír mint a szallas.hu.

A szallas.hu-hoz hasonlóan ezen a portálon is részletesen lehet szűrni a szálláshelyek tulajdonságait. A találatok között szintén a szálláshelyek jelennek meg, amelyeknek részletes leírásában tekinthetők meg a szobák. A foglalásban csak egy szálláshely szobái szerepelhetnek. A keresés során maximálisan 30 felnőtt és 10 gyerek választható.

A portál rendelkezik értékelési rendszerrel.

## Trivago.hu

A trivago.hu a Trivago nemzetközi szálláskereső szolgáltatás Magyarországra készült változata. A működése eltér az 2.1 és 2.2 fejezetekben tárgyalt portálokétól, ugyanis a Trivago csak összegyűjti más szálláskereső portálok ajánlatait és azok közül keres.

A keresési feltételekkel nagyvonalúan bánik, nem lehet elég részletesen beállítani a kívánalmakat. Lehet szűrni a teljes foglalás ára és a talált szálláshelyek városközponttól számított távolsága alapján. Az előző fejezetekben megvizsgált portálokhoz hasonlóan ez a rendszer sem képes a szobákat vegyesen ajánlani. A keresési találatok mindig egy-egy szálláshelyre vonatkoznak. A csoportos szálláskeresést csak korlátozottan támogatja. Az utazó személyek kiválasztásakor maximum 16 felnőtt és 16 gyerek választható.

## Konklúzió

A magyar szálláskereső piac portáljai jó felületet nyújtanak az egyéni utazók számára. A vizsgált portálok előnyben részesítik a szálláshelyeket és jellemzően egy szálláshelyre koncentrálják ajánlataikat. A keresési szempontokat mindhárom portál esetében kielégítőnek találtam. Az utazó személyek száma a keresés során mindenütt korlátozott. Egyik portál sem képes több szálláshelyről származó szobákat egy foglalásként kezelni.

A kutatásom során nem találtam olyan szálláskereső portált, amely funkcionalitásában közvetlen vetélytársa vagy alternatívája lehetne az általam felvázolt rendszernek.

# Nemlineáris programozás

# Ruby on Rails

# Specifikáció

A feladat teljesítéséhez egy webalkalmazás tervezése és implementálása volt a cél. A webes technológia választásának oka, hogy a már megszokott és ismert szálláskereső portálokhoz hasonuljon. Ezen kívül az internetes platformra való fejlesztéssel lehet a legolcsóbban és leggyorsabban a legszélesebb felhasználói kört elérni. A manapság rendelkezésre álló úgynevezett *responsive*, magyarul alkalmazkodó web design-ok alkalmassá tesznek egy weboldalt arra, hogy egyszerre legyen áttekinthető és kezelhető minden képernyőméreten.

## Funkcionális követelmények

A fejezet a webalkalmazással szemben támasztott követelményeket és elvárásokat taglalja.

### Szereplők

A tervezett rendszerben négy felhasználói szerepkör különül el, amelyek a következők:

1. **Látogató**: bejelentkezés nélkül böngészi a portál publikus tartalmát.
2. **Szálláskereső**: bejelentkezés után szobát keres és foglal
3. **Szállásadó**: bejelentkezés után szobákat hirdet, foglalásokat kezel
4. **Adminisztrátor**: bejelentkezés után a rendszer törzsadatait és beállításait kezeli

### Szobák szűrése

A látogatónak és a szálláskeresőnek lehetőséget kell biztosítani a szobák szűrésére. A szűrési feltételek között szerepelnie kell a szálláshely szolgáltatásainak, a szoba felszereltségének, a szoba elérhetőségét jelző kezdő- és végdátumnak, a szoba típusát jelző ágyak számának valamint a városnak.

### Szobafoglalás

A szálláskereső csak a kiválasztott időszakban a rendszer által elérhetőként nyilvántartott szobákat foglalhatja le. A foglalás véglegesítése előtt a szálláskeresőnek minden vendég adatát meg kell adnia.

A szobafoglalásról minden szállásadónak egyénileg kell visszajelzést készítenie. A szobafoglalást el lehet fogadni és vissza lehet utasítani. Egy foglalás akkor tekinthető teljesíthetőnek, ha minden szállásadó pozitív visszajelzést küldött. A foglalás nem teljesíthető, ha legalább egy szállásadó negatív visszajelzést küldött.

A szobafoglalások a szálláskereső és a szállásadó részéről is bármikor visszakereshetők és megtekinthetők.

### Értékelés

A teljesült szobafoglalások esetén, az utazás befejező dátumát követően a szálláskereső értékelheti a meglátogatott szálláshelyeket.

### Intelligens keresés

Az intelligens keresés funkció ár és távolság, vagy ezek kombinációja szerint képes automatikus ajánlást készíteni. A választható szempontok mellett figyelembe kell vennie a szálláshelyek értékeléseit és törekednie kell a jobb értékelésűek ajánlására.

### Törzsadatok

A rendszerben megjelenő törzsadatokat az adminisztrátornak szerkesztenie és bővítenie kell tudnia.

### Tartós címek

A rendszerben megjelenő oldalak címeit és a keresések eredményoldalaira mutató címeket úgy kell kialakítani, hogy azok bármikor újra meglátogathatóak és linkelhetőek legyenek.

## Célcsoport

A webalkalmazás felhasználói célcsoportjaként a szállásadó szerepkör részéről a jellemzően vidéki, alacsony kapacitású panziókat és apartmanokat azonosítottam. Számukra a rendszer ugyanúgy a foglalások egyszerű kezelhetőségét nyújtja, mint a szálláskeresők számára. A szálláskereső szerepkör szempontjából a célcsoport tagjaiként az iskolai kirándulásokat szervező osztályfőnök, a baráti társaságok, illetve az üzleti célból szállást kereső szervezőket tekintem.

# Tervezés

A fejezet a feladat megvalósításához szükséges tervezés eredményét mutatja be. A fejezet kitér az alkalmazásban megjelenő folyamatok tárgyalására, bemutatja az intelligens keresés működéséhez szükséges optimalizációs modelleket. A fejezet második felében a tervezett adatbázis entitásai és a megvalósítás során felhasznált technológiákról lesz szó.

## A rendszerben megjelenő fő folyamatok

Ez a fejezet a rendszerben megjelenő fő interakciós és háttérfolyamatokat mutatja be.

### Szobafoglalás

A szobafoglalás folyamatában a bejelentkezett szálláskereső valamelyik keresési mechanizmust választva feltölti a virtuális kosarát a foglalni kívánt szobákkal. A kosár feltöltése után a szálláskereső véglegesíti a foglalását, megadja a foglalásban részt vevő vendégek adatait és a foglalást elküldi. Az alábbi ábra a folyamat lépéseit részeltesen mutatja be.



. ábra Szobafoglalás folyamata

### Foglalás visszaigazolás

A rendszerbe érkező szobafoglalásokat a szállásadóknak külön-külön vissza kell igazolniuk. A foglalás állapota csak akkor változhat meg, ha minden szállásadó megtette visszajelzését. Az alábbi ábra bemutatja a visszaigazolás folyamatát a foglalás szempontjából.



. ábra Foglalás visszaigazolás folyamata

Miután minden szállásadó visszaigazolta a rá vonatkozó szobákat, a rendszer új állapotba lépteti a foglalást. A foglalás teljesíthető állapotúvá válik, ha minden szállásadó pozitív visszajelzést adott. A foglalás nem teljesíthető állapotú lesz, ha legalább egy szállásadó negatív választ adott.

### Intelligens keresés

Az intelligens keresés háttérfolyamatát a rendszer a szálláskereső által megadott keresési feltételek alapján végzi el. A folyamat lépéseit részletezi az alábbi ábra.



. ábra Intelligens keresés háttérfolyamata

A rendszer nem tud közvetlenül kommunikálni a nemlineáris megoldóval ezért előbb a szűrési feltételek szerint kiválogatott szobák alapján elkészíti az optimalizációhoz szükséges adatmodellt és azt, az adatbázisból kiolvasott modellel együtt fájlba írja. Ezután parancssorból végzi a nemlineáris megoldó futását és az eredmények kiolvasását.

## Nemlineáris programozási modell

A következőkben az intelligens keresés funkcióhoz használt nemlineáris programozási modelleket mutatom be.

A nemlineáris optimalizáció során a cél az, hogy ár, távolság, illetve minőség szempontjából optimális megoldást kell találni. A minőség, vagyis a szobák a szálláshelytől örökölt értékelése minden modellben megjelenik, hiszen cél az is, hogy a szálláskereső számára nem csak racionálisan, de emocionálisan is elfogadható megoldást kínáljon a rendszer. Az ár és a távolság választható külön-külön és együttesen is. Tehát három különféle modellt kellett kialakítanom.

A modellek kialakítása során figyelembe kellett vennem, hogy a különböző szempontokhoz különböző nagyságrendű és szórású értékek tartoznak. Az ár jellemzően tízezres nagyságrendű érték. A távolság, amennyiben a keresés egy városra terjed ki a pár tíz kilométernél nem nagyobb, míg város meghatározása nélkül több száz kilométer is lehet. Az értékelés egy 1-től 10-ig terjedő skálán számított átlagos érték. A nemlineáris modellben a célfüggvény a kifejezés minimalizálására törekszik. Ezáltal belátható, hogy a nagyobb nagyságrendű értékektől fog függni a megoldás. Ez nem megfelelő, a megoldás szempontjából minden szempontnak egyenlően kell teljesülnie.

A különböző nagyságrendű értékeket két módszerrel tettem összehasonlíthatóvá. Az első módszerem az, hogy az ár és távolság értékeket nem közvetlenül használom fel az adatmodellben. Az adatmodellbe jegyzés előtt növekvő sorrendbe állítom őket, és minden különböző értéket 1-től egyesével növelve kategóriákba sorolom, ahogy az alábbi ábrákon is látható.



. ábra Árak kategorizálása (Ft)



. ábra Távolságok kategorizálása (km)

Az ár és távolság értékekből annyi kategóriát különböztetek meg, ahány különböző érték megjelenik a kiértékelés során. Mivel ez jellemzően nem haladja meg a 20-30-as számosságot, ezért az értékelésekkel is jobban összevethető. Hangsúlyos előny továbbá, hogy megszűnik a sokaság gyakran előforduló kiugró szórása, ami a következő módszer előnyére is válik.

A fenti módszerrel kialakított kategóriák legnagyobb értéke akár a duplájával is meghaladhatja az értékelések legnagyobb, 10 értékét, azonban a sokaságok szórása közel hasonló értékekkel bír. A célfüggvényben tehát úgy döntöttem, hogy nem a puszta összegeket tekintem, hanem változók által kijelölt ár- és távolságkategóriák, valamint az értékelések sokaságainak speciális relatív szórását. A relatív szórás azért speciális, mert nem a középértékhez közelítem, hanem az ár- és távolságkategóriák esetében a legkisebb, 1 értékhez, míg az értékelések esetében, a legnagyobb 10 értékhez. A relatív szórás eredménye egy százalékos szám. A célfüggvény tehát három százalékérték összegét minimalizálja. Az alábbi képlet az alkalmazott relatív szórási képletet mutatja be, ahol *si* a vizsgált sokaság egy értéke, *xi* a bináris súly, *smin* a vizsgált sokaság lehetséges legkisebb értéke.

6. képlet Speciális relatív szórás képlet

A fenti módszerekkel el tudtam érni, hogy több, különböző nagyságrendű sokaságot összehasonlítsak és a célfüggvény kiértékelésekor az algoritmus azokat egyenlőként kezelje.

A *6.1* ábrán bemutatott képlet miatt szükséges, hogy az optimalizációt nemlineáris megoldóval végezze a rendszer. A linearitást a bináris súllyal – ami változóként szerepel a modellben – való szorzással lépi át a modell.

Mindegyik modell bináris változókat használ, amik azt mutatják, hogy mely szobákat kell a megoldáshalmazba beválasztani. A modellekhez alapvetően két adathalmazra van szükség. Az első a szobák adathalmaza, amely minden eleméhez legalább kettő paraméter tartozik: kapacitás és értékelése. A második adathalmaz a vendégek száma. Az optimalizálási szempontok szerint a szobák további paraméterekkel bővülnek. Az alábbi ábra a szobák halmazának egy elemét és a hozzá kapcsolódó változót és paramétereket mutatja be.



. ábra A modellben megjelenő szoba objektum és a hozzá kapcsolódó változó és paraméterek

Mindhárom modellben egy alapvető korlátozást vezettem be, egyértelmű módon azt, hogy a kiválasztott szobák kapacitása egyenlő kell, hogy legyen a vendégek számával.

6. képlet Korlátozás a vendégek száma alapján

A fenti képletben *fi* bináris változó, *ci* az *i*-edik szoba kapacitása, *v* pedig a vendégek száma.

### Olcsó modell

Az olcsó modell azokat a szobákat adja eredményül, amelyek a legolcsóbbak és a lehető legmagasabb értékeléssel bírnak.



. ábra Az olcsó modellhez szükséges paraméterek

Ahogy azt a fenti ábra is mutatja, ehhez a modellhez a szoba halmaz paraméterlistáját ki kell egészíteni az szobák árával.

. képlet Az olcsó modell célfüggvénye

A modell célfüggvényét a fenti ábra mutatja be, ahol *pi* az i-edik szoba ára, *pmin* a legalacsonyabb szobaár a sokaságban, *ri* az i-edik szoba átlagos értékelése *rmax* pedig a lehetséges legmagasabb értékelés.

### Közeli modell

A közeli modell azokat a szobákat választja ki, amelyek egymáshoz képest a legközelebb helyezkednek el és a lehető legmagasabb értékeléssel bírnak.



. ábra A közeli modellhez szükséges paraméterek

A távolságok tárolásához egy, a szobák halmazán képzett Descartes szorzatból kialakított mátrixra van szükség, ahol a távolság paraméterként jelenik meg, ahogy az a fenti ábrán is látható. A közös szálláshelyen lévő szobák távolsága 0.

. képlet A közeli modell célfüggvénye

A modell célfüggvényét a következő ábra mutatja be, ahol *dij* az i-edik és j-edik szoba távolsága, *dmin* a legalacsonyabb távolság a sokaságban, *ri* az i-edik szoba átlagos értékelése *rmax* pedig a lehetséges legmagasabb értékelés.

### Olcsó és közeli modell

Az olcsó és közeli modell egyesíti a 6.2.1 és 6.2.2 fejezetekben taglalt modelleket, vagyis az egymáshoz legközelebb eső legolcsóbb és lehető legmagasabb értékeléssel bíró szobákat adja eredményül.



. ábra Az olcsó és közeli modellhez szükséges paraméterek

Az összevont modellnek szüksége van minden, az előző két fejezetben tárgyalt kiegészítő paraméterre, ahogy az a fenti ábrán is látható.

. Az olcsó és közeli modell célfüggvénye

Az egyesített célfüggvényt mutatja a fenti ábra, ahol *dij* az i-edik és j-edik szoba távolsága, *dmin* a legalacsonyabb távolság a sokaságban, *pi* az i-edik szoba ára, *pmin* a legalacsonyabb szobaár a sokaságban *ri* az i-edik szoba átlagos értékelése *rmax* pedig a lehetséges legmagasabb értékelés.

## Adatbázis tervezet

## Technológia

### Ruby on Rails

### PostgreSQL

### AMPL

### Bonmin

### HTML, CSS, Javascript

### Gems…

# Megvalósítás

## Autentikáció és autorizáció

## Szobák szűrése

## Intelligens keresés

## Szobafoglalás

# Felületek és használat

## Menüsáv

## Szobák

## Szálláshelyek

## Foglalások

## Intelligens keresés

## Kosár

## Adminisztrációs felületek

# Tesztelés

## Tesztelési környezet

## Teszt adatok

## Teszt eredmények

# Összefoglalás

Irodalomjegyzék

Folyóirat cikk:

NASH, L., SMIDTH, G. (1999). The Alpha-Clustering. *Journal of Computing*. 5(2): 17-29.

Könyv:

KIMT, G. (1998). *A Fuzzy Logic Method In Window Design*. Springer, Berlin Heidelberg New York.

Fejezet (könyvben vagy proceedings-ben):

HINTON, H. (1997). The Heavens are Falling. In: Rosenberg, K. (ed.): *Random Thoughts*. MIT Press, 40-100.

Internetes hivatkozás (CD-n beadnadó)

<http://mik.uni-pannon.hu/index.php?func=news&main=262> MÉSZÁROS P. (letöltés dátuma 2008. október 10.) *Záróvizsga információk 2009. január*

Aláhúzás nélkül kell megadni az internet címeket.

Mellékletek

CD Melléklet

dolgozat (pdf-ben és az eredeti szerkeszthető formában is), internetes hivatkozások letöltött anyagai, összes elkészített saját munka (pl programkód, fénykép stb.)