Szoftverarchitektúrák   
házi feladat  
  
Közösségi Parkolási portál  
  
*németh Balázs, KSPG27  
Nacsa Gábor, HB6463  
  
konzulens: Sipos márton*

Table of Contents

[2 Specifikáció - 3 -](#_Toc404717954)

[2.1 Fejlesztői csapat: - 3 -](#_Toc404717955)

[2.2 Részletes feladatleírás: - 3 -](#_Toc404717956)

[2.3 Technikai paraméterek - 5 -](#_Toc404717957)

[2.4 Use-case diagram: - 5 -](#_Toc404717958)

[3 A rendszer - 6 -](#_Toc404717959)

[3.1 A rendszer környezete - 6 -](#_Toc404717960)

[4 Adatmodell - 7 -](#_Toc404717961)

[5 Szoftverarchitektúra - 7 -](#_Toc404717962)

[6 Felhasználói felület - 7 -](#_Toc404717963)

[7 References - 8 -](#_Toc404717964)

# Specifikáció

A hallgatók feladata egy olyan térkép alapú online rendszer kifejlesztése, amelyik lehetővé teszi felhasználóinak, hogy megosszák az általuk ismert  ingyenes/előnyös parkolási lehetőségeket a többi felhasználóval. A rendszer tegye lehetővé fényképek feltöltését és a cím alapú keresést. Legyen elérhető mobil készülékekről.

## Fejlesztői csapat:

|  |  |  |
| --- | --- | --- |
| **Név** | **Neptun kód** | **E-mail cím** |
| Németh Balázs | KSPG27 | [nem841425bal@gmail.com](mailto:nem841425bal@gmail.com) |
| Nacsa Gábor | HB6463 | [nacsa.gbr@gmail.com](mailto:nacsa.gbr@gmail.com) |

A csapatban szerepek kiosztását a csapat kis mérete miatt nem tartottuk fontosnak. Az effektív csapatmunka érdekében létrehoztunk GitHubon egy nyilvános projektet, mely segítségével verziókövetés mellett végeztük a fejlesztés egészét [1].

## Részletes feladatleírás:

Célunk egy olyan webes alkalmazás készítése, melynek felhasználói menedzselhetik és megoszthatják egymással parkolóhelyeiket:

* Parkolóhely felvétele: A parkolóhely címének, árának, elérhetőségi sávjainak megadásával.
* Kép felvétele parkolóhelyről: A parkolókhoz opcionálisan fényképek is társíthatók a könnyebb megtalálás érdekében.
* Parkolóhelyek megjelenítése térképen: A parkolóhelyek böngészhetők az adott terület térképén való megjelenítésével, hogy a felhasználó a számára legkedvezőbb helyet találhassa járművének.

Parkolóhelyek szűrése: A parkolóhelyek szűrése az alábbi szempontok alapján végezhető, tetszőleges kombinációban:

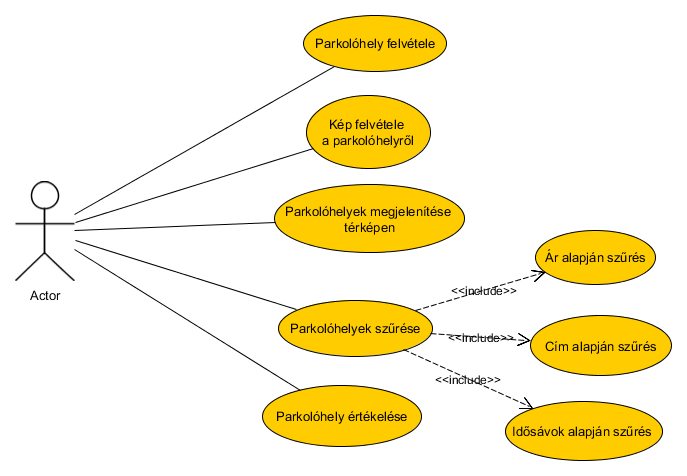
* + Ár alapján: Megadható felső árkorlát, így ingyenesre is szűrhetünk.
  + Cím alapján: Megadható, hogy legfeljebb mekkora távolságra lévő parkolóhelyeket szeretnénk megjeleníteni az adott címtől.
  + Idősávok alapján: Lekérhető, hogy a kívánt idősávban mely parkolók érhetők el.
* Parkolóhely értékelése: A felhasználók minősíthetik az egyes parkolóhelyeket, hogy mennyire találták azt összességében jónak, ezzel segítve a többi felhasználót.

Az alkalmazás implementálásához elengedhetetlen felhasználó kezelés megvalósítása. A felhasználók regisztrálhatnak, majd később ezzel a profillal vehetik igénybe az alkalmazás nyújtotta szolgáltatásokat.

## Technikai paraméterek

Vékonykliens alkalmazásunk egy webszerveren fog futni, így webböngészőből lesz elérhető a felhasználók számára, ezért a klienstől kizárólag egy webbrowser meglétét követeljük meg.

## Use-case diagram:



# A rendszer

Az elkészített szoftver egy webes alkalmazás, mely lehetővé teszi felhasználóinak, hogy egy nyilvános rendszerbe felvegyenek parkolóhelyeket, és megoszthassák egymással ezekről alkotott véleményeiket, értékelésüket, tapasztalataikat.

A felhasználók regisztrálhatnak a rendszerbe, szerkeszthetik egymás által felvett parkolóhelyeket, jelölhetik őket Google Maps térképen és tölthetnek fel fényképeket a helyszínekről. Valamint lehetőségük van több szempont alapján szűrni a közösség által felhalmozott tudásbázisban.

## A rendszer környezete

Alkalmazásunk futtatásához csak egy webböngésző kell JavaScript támogatással, így gyakorlatilag minden operációs rendszerről és minden böngészőből gond nélkül elérhető, mobil készülékről is.

A szoftver fejlesztése Java nyelven történt a Vaadin [2] keretrendszer 7.3.3-as verziójának segítségével, mely biztosítja számunkra a könyvtárakat a rugalmas felhasználói felület elkészítésére.

Az adatbázis rendszerünk megvalósításhoz az SQLite [3] köztulajdonban lévő (public domain), nyílt forráskódú könyvtárat használtuk, mely egy pehelysúlyú, SQL alapú adatbázis kezelő rendszer, mely nem igényel különálló infrastruktúrát az üzemeltetéséhez.

# Adatmodell

Az adatmodell tervezésénél különös figyelmet fordítottunk a rendszer kiterjeszthetőségére, illetve arra, hogy minél kevesebb redundanciával és hatékonyan tudjuk tárolni az adatokat.

Először megállapítottuk, hogy mit szeretnék gyűjteni a parkolóhelyekről, ahhoz hogy minden követelmények meg tudjon felelni az alkalmazásunk és sokféle felhasználási lehetőséget biztosítson. Ezeket figyelembe véve az alábbi attribútumokat gyűjtöttük össze (zárójelben az adattáblában, illetve a programban használt elnevezések láthatóak) :

* Parkoló címe (address)
* GPS koordinátái (lat, lon)
* A parkolás ára (price)
* Hány órától vehető igénybe (availfrom)
* Hány óráig vehető igénybe (availuntil)
* Ki vette fel az adatbázisba (username)
* Képek a parkoló környékéről, helyéről (picture)
* Értékelés (rating)
* Kommentár (comment)
* Regisztrált felhasználók felhasználónevei (username)
* A felhasználók jelszavai (password)

Az adatbázist három táblába dekomponáltuk:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id | username | lat | lon | address | price | availfrom | availuntil |

Tábla 1: A parking nevű adatbázistábla oszlopai

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | username | picture | rating | comment |

Tábla 2: A parlrating nevű adatábzistábla oszlopai

|  |  |
| --- | --- |
| username | password |

Tábla 3: A users tábla oszlopai

Az adatbázison értelmezett funkcionális függések az alábbiak:

* id -> {username, lat, lon, address, price, availfrom, availuntil}
* {id, username} -> {picture, rating, comment}
* username -> password

BCNF-e

A *parking* tábla elsődleges kulcsa az azonosító (*id*) oszlop, melyet az adatbázis generál az adatrekord felvételekor. Az automatikus generálást az SQLite *AUTOINCREMENT* kulcsszavával valósítottuk meg, így a kulcs egyedisége is rögtön biztosított. A *username* oszlop idegen kulcs a *users* táblából, értéke tetszőleges alfanumerikus karakterlánc lehet, melyet a felhasználó a regisztrálás során választ, ebben a táblában azt a felhasználót azonosítja, aki felvette a parkolóhelyet a rendszerbe. A *lat* és *lon* oszlopok előjeles valós számként tárolják a parkolóhely GPS koordinátáit.

Adattáblák, NF, mit tesz lehetővé, milyen constraintek

# Szoftverarchitektúra

Osztálydiagram, architektúra

# Felhasználói felület

Felhasználói felület bemutása, esetleg egykét példa

# References

|  |  |
| --- | --- |
| [1] | Németh Balázs, Nacsa Gábor, "Közösségi Parkolási Portál," https://github.com/nacsa/szoftarch. [Online]. |
| [2] | Joonas Lehtinen, CEO, "Vaadin," vaadin.com. [Online]. |
| [3] | Public Domain, "SQLite - Small. Fast. Reliable. Choose any three.," http://www.sqlite.org/. [Online]. |