**EcoMap**

Specifikáció és Rendszerterv

http://www.adatvilag.hu/ecomap

Készítette: Werner Zoltán

A FejlesztőLány blog ötlete alapján: <https://fejlesztolany.hu/otletek/>

**A fejlesztés célja**

A fejlesztés célja egy olyan térképi alapú alkalmazás, amely a környezetvédelemért elkötelezett felhasználók számára nyújtja a következő szolgáltatásokat:

1. Térképi és listaszerű megjelenítése a következő helyszíneknek

* szelektív kukák
* komposztpontok
* csomagolásmentes üzletek
* elektronikai és veszélyes hulladék leadó pontok
* javítóműhelyek

1. Környezetvédelmi események megjelenítése térképen és listában

Az EcoMap lehetővé teszi a felhasználók regisztrációját követően új pontok és események felvitelét és a meglévők módosítását és törlését előzetes adminisztrátori jóváhagyást követően.

A felhasználó aktuális lokációja alapján továbbá megjelenítésre kerülnek a 10 km-en belül található elemek.

**A fejlesztés technikai keretei**

Az EcoMap fejlesztése során az alábbi technikai rétegek kialakítása történik meg:

* **Adatbázis réteg:** az EcoMap felületén megjelenítendő adatkörök tárolását, valamint a térinformatikai megjelenítéshez szükséges geometriai adatok tárolására szolgál. Az adatbázisban kerülnek továbbá tárolásra a bejelentkezési és jogosultságokat tartalmazó adatok. A felhasználói adatok titkosítva kerülnek tárolásra. A fejlesztés során **PostgreSQL** szervert használunk.
* **Üzleti logikai réteg:** az adatbázis és a felhasználói megjelenítő réteg közötti az adatfeldolgozás, valamint az üzleti logika megvalósítására **Java Spring Boot** keretrendszert használunk. **Spring Security** használatával az EcoMap rendszer biztonsági követelményei is itt kerülnek megvalósításra.

Itt történik továbbá az automatikus levélküldés kifejlesztése az adminsztrátorok felé, ha egy felhasználó egy eseményt vagy helysínt módosított vagy vitt föl vagy törölt.

* **Felhasználói megjelenítő réteg:** a felhasználók számára térképi felületen történik az adatok megjelenítése, de emellett listaszerű lekérdezés, előzetes regisztrációt követően módosítási lehetőségre szolgáló felület is kifejlesztésre kerül. A térképi felületet az OpenStreetMap API-n keresztül biztosítjuk. A felhasználói megjelenítés **Angular** keretrendszerrel kerül kifejlesztésre.

**Rendszerterv**

**PostgreSQL adatbázis:**

Az EcoMap adatbázis rétegében a következő adattáblák kialakítása történik meg:

* Felhasználók és jogosultságok tárolása:
  + USERS: a regisztrált felhasználók adatait tartalmazó tábla. Mezői:
    - id
    - username
    - password (pgcrypto alkalmazásával titkosítottan)
    - registered (a regisztráció elfogadásra került: 1, a regisztráció nem került elfogadásra: 0)
  + ROLES: a kiosztható jogosultságokat tartalmazó tábla. Mezői:
    - id
    - role\_name:
      * ADMIN (adminisztrátor)
      * ADD\_EVENTS (események módosítási joga)
      * ADD\_LOCATIONS (helyszínek módosítási joga)
  + USERS\_ROLES: kapcsolótábla a USERS és az ROLES tábla között, mivel egy felhasználó több jogosultsággal is rendelkezhet. Mezői:
    - users\_id
    - roles\_id
* Környezetvédelmi helyszínek tárolása:
  + LOCATIONS: a helyszínek adatait tartalmazó adattábla. Mezői:
    - ID
    - TYPE: a helyszín típusa. Lehetséges értékei:
      * TRASH: szelektív kuka
      * COMPOST: komposztpont
      * SHOP: csomagolásmentes üzlet
      * ELECTRONIC: elektronikai és veszélyes hulladék pont
      * REPAIR: javítóműhely
    - NAME: a helyszín megnevezése
    - ADDRESS\_CITY: város megnevezése
    - ADDRESS\_ROAD: útnév
    - ADDRESS\_NUM: házszám
    - COORD\_X: WGS X koordináta
    - COORD\_Y: WGS Y koordináta
    - GEOM: geometriát tartalmazó mező
    - DATE: rögzítés dátuma
    - USER\_ID: rögzítő ID-ja
    - MAKE\_DEL: törlési igény esetén értéke 0
    - IS\_ALIVE: létező (1) vagy nem létező (0) rekord
* Környezetvédelmi események tárolása:
  + EVENTS: az események adatait tartalmazó adattábla. Mezői:
    - ID
    - TYPE: az esemény típusa (EVENT)
    - NAME: a helyszín megnevezése
    - DATE\_FROM: az esemény kezdő dátuma
    - DATE\_TO: az esemény vége dátuma
    - ADDRESS\_CITY: város megnevezése
    - ADDRESS\_ROAD: útnév
    - ADDRESS\_NUM: házszám
    - COORD\_X: WGS X koordináta
    - COORD\_Y: WGS Y koordináta
    - GEOM: geometriát tartalmazó mező
    - DATE: rögzítés dátuma
    - USER\_ID: rögzítő ID-ja
    - MAKE\_DEL: törlési igény esetén értéke 0
    - IS\_ALIVE: létező (1) vagy nem létező (0) rekord. Automatiksan 0-ra vált, ha a DATE\_TO értéke magasabb mint a DATE\_NOW() értéke

**Java Spring Boot:**

Az üzleti logikát, valamint az adatbázis és a felhasználói réteg közötti adatfeldolgozást Java Spring Boot keretrendszerben valósítjuk meg. Itt történik továbbá Spring Security keretében a biztonsági keretek megvalósítása is.

1. **Repository réteg:**

* JPA interfészen keresztül történik az adattáblák elérése
* felhasználói jogosultságtól függően az egyes adattáblákhoz a következő hozzáférések lesznek:
  + SELECT jog a LOCATIONS és EVENTS táblára: minden, bejelentkezés nélküli felhasználó számára
  + UPDATE, INSTERT jog a LOCATIONS táblára: minden bejelentkezett, ADD\_LOCATIONS joggal rendelkező felhasználó számára, de:
    - az IS\_ALIVE értékét csak 0-ra állíthatja új rekord felvitele esetén
    - az IS\_ALIVE értékét nem állíthatja 0-ra, csak a MAKE\_DEL értékét

(…)

1. **Service réteg**

* email küldés

1. **Controller réteg**
2. **Security réteg**

**View réteg (Angular):**

* a view réteg megvalósítása Angular keretrendszerrel történik meg
* két fő részre osztható a weboldal:
  + OSM API alapú térképi felület, alapértelmezett, lásd utinform.hu, itt is lehet új objektumot felvinni és meglévőt módosítani.
    - Keresési lehetőség legyen
    - Keresés utcanév, házszámra is
  + táblázatos felület, külön menüpontra, itt utcanév, házszám alapján lehet újat felvenni
    - Keresési llehetőség legyen itt is

<https://blog.softwaremill.com/openstreetmap-in-house-alternative-to-google-maps-8097a087cc66>

<https://medium.com/@balramchavan/using-openstreetmap-inside-angular-v6-3d42cbf03e57>

<https://mediarealm.com.au/articles/openstreetmap-openlayers-map-markers/>

JSON:

<https://medium.com/@bcarunmail/using-json-datatype-in-relational-database-to-develop-flexible-configurable-software-6a169e9ce75b>

<https://www.baeldung.com/spring-boot-json>