**Documentul de Proiectare a Soluţiei Aplicaţiei Software**

(Software Design Document)

Version 1.0

8 Decembrie, 2017

**Ingineria Sistemelor Software**

Water Monitoring

Facultatea de Automatică şi Calculatoare, Universitatea Politehnica, Bucureşti

Beboeanu Mihaela-Elena 343C5

Coriiu Iustinian-David 343C5

Groapa Mihai 343C5

Cuprins

[1. Scopul documentului 3](#_Toc503478820)

[2. Obiective 3](#_Toc503478821)

[3. Continutul documentului 3](#_Toc503478822)

[4. Use Cases 4](#_Toc503478823)

[4.1 Diagrama Use Case 4](#_Toc503478824)

[4.2 Creare cont 4](#_Toc503478825)

[4.3 Schimbare Parola 4](#_Toc503478826)

[4.4 Autentificare 4](#_Toc503478827)

[4.5 Lucrul cu harta autentificat ca “Voluntar” 4](#_Toc503478828)

[4.6 Raportare incident 5](#_Toc503478829)

[4.7 Stergere incident 5](#_Toc503478830)

[4.8 Vizualizare incident 5](#_Toc503478831)

[4.9 Lucrul cu harta autentificat ca “Authority”/Entitate legala 5](#_Toc503478832)

[4.10 Adaugare senzor 5](#_Toc503478833)

[4.11 Stergere sensor 5](#_Toc503478834)

[4.12 Vizualizare date masurate de sensor 6](#_Toc503478835)

[4.13 Vizualizare grafic cu probele luate de un sensor 6](#_Toc503478836)

[4.14 Logout 6](#_Toc503478837)

[5. Arhitectura sistemului 7](#_Toc503478838)

[5.1 Diagrama arhitecturii de sistem 7](#_Toc503478839)

[5.2 Serverul 7](#_Toc503478840)

[5.3 Clientul 7](#_Toc503478841)

[6. Descrierea componentelor 9](#_Toc503478842)

[6.1 Diagrama de module 9](#_Toc503478843)

[6.2 Modulul server 9](#_Toc503478844)

[6.3 Modulul client 9](#_Toc503478845)

[6.3.1 Modulul api 10](#_Toc503478846)

[6.3.2 Modulul appl 10](#_Toc503478847)

[6.3.3 Modulul Mail 10](#_Toc503478848)

[6.3.4 Modulul Autentificare 11](#_Toc503478849)

[6.3.5 Modulul GUI 11](#_Toc503478850)

[6.3.6 Modulul Picture 11](#_Toc503478851)

[7. Baza de date 11](#_Toc503478852)

[7.1 Diagrama bazei de date 11](#_Toc503478853)

[7.2 Tabelele bazei de date 13](#_Toc503478854)

[8. Tehnologii folosite 14](#_Toc503478855)

[9. Riscuri ce impiedica terminarea aplicatiei 14](#_Toc503478856)

[10. Metode de a minimaliza riscurile 14](#_Toc503478857)

# 1. Scopul documentului

Scopul acestui document este de a descrie modul in care se desfasoara implementarea aplicatiei in ArcGIS. Aplicatia este implementata astfel incat sa contureze o platforma placuta si relaxanta, cu o interfata creativa, care sa mapeze resursele de apa dintr-o zona a Romaniei si sa o monitorizeze printr-o retea de senzori.

# 2. Obiective

* Modelarea sistemului de monitorizare al resurselor de apa prin simularea entitatilor implicate in acest proces de preventie (resursa de apa, senzori de monitorizare, date receptionate, entitati legale, voluntari, etc.);
* Modelarea relatiilor dintre entitatile implicate in procesul de monitorizare;
* Proiectarea si implementarea unei componente care afiseaza grafice privind calitatea apei în functie de resursa de apa monitorizata si care trimite alerte (email, SMS) in caz de poluare sau valori prea mari ale anumitor parametri;
* Proiectarea si implementarea unei componente care permite voluntarilor sa raporteze anumite incidente legate de resursele de apa.

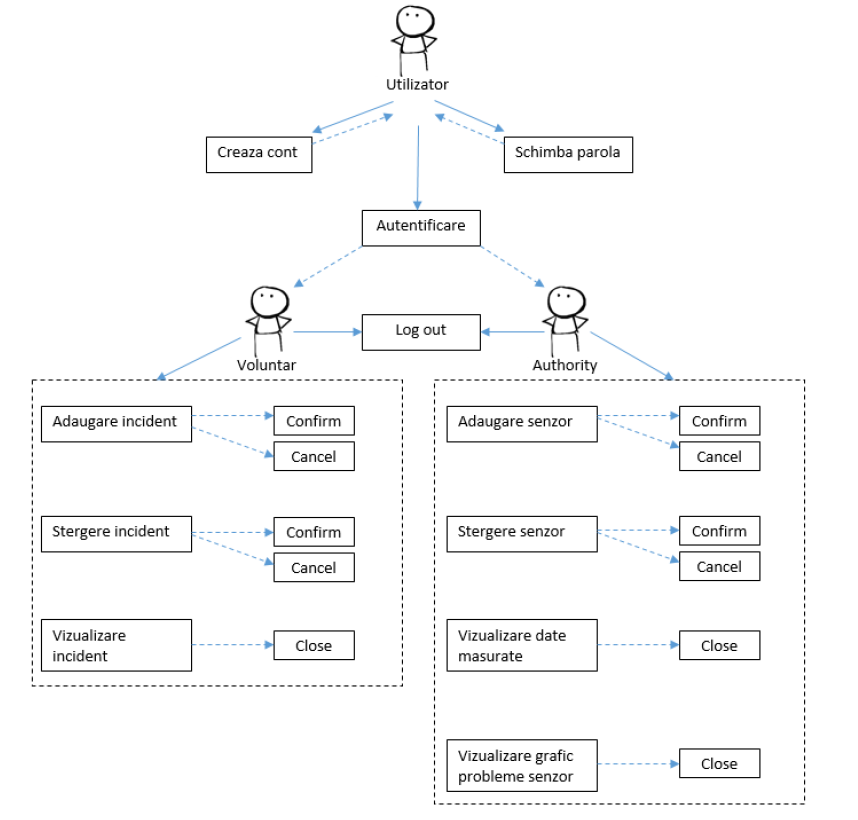
# 3. Continutul documentului

Documentul are scopul de a transpune modul in care va fi construit sistemul software astfel incat sa aiba comportamentul descris in documentul ‘Specificarea Cerintelor Software’. De asemenea, acesta prezinta metodologii, tehnologii, participanti si resurse implicate in mod direct in proiect.

* *Use Cases* – abordarea schematica a proiectului
* *Arhitectura Sistemului* - prezinta interfata cu utilizatorul contribuie la construirea unei imagini detaliate a modului in care este implementat proiectul
* *Descrierea Componentelor* – descrierea modulelor si submodulelor ce formeaza aplicatia
* *Descrierea Bazei de Date*
* *Riscuri ce impiedica terminarea aplicatiei*
* *Tehnologii folosite*

# 4. Use Cases

## 4.1 Diagrama Use Case



## 4.2 Creare cont

* Utilizatorul deschide aplicatia
* Apasa pe butonul „Not a member? Join now for free”
* Sitemul afiseaza fereastra de creare cont, cuprinzand: label pentru fotografia de profil, buton de browsing pentru a incarca poza, campuri de tip text pentru nume, prenume, email, un date picker pentru data nasterii, un dropdown din care se poate alege tipul contului, avand doua optiuni: „Authority” si „Volunteer” si nu in ultimul rand, doua campuri pentru parola si reintroducere parola. De asemena, vor exista butoane pentru clear (aduce toate elementele la starea initiala), back (afiseaza fereastra de login) si submit, pentru a trimite catre server cerere de creare a contului.
* Cazuri limita, tratate de sitem si marcate prin afisarea unor mesaje de eroare: email-ul introdus nu este valid, email-ul este asociat deja unui alt cont, parola introdusa contine caractere nepermise (spatii, tab-uri, simboluri din afara ariei cuprinse de o tastatura in limba engleza), parola este prea scurta, parola si parola „retyped” difera.

## 4.3 Schimbare Parola

* Utilizatorul deschide aplicatia
* Apasa pe butnoul „Change password”
* Sistemul afiseaza fereastra de resetare parola, compusa din: camp pentru introducerea mail-ului, buton de trimitere cod de resetare pe email, camp pentru introducerea codului primit, doua campuri de tip password pentru parola. De asemena, are butoane pentru anularea cererii de schimbare parola si revenirea la fereastra de login, precum si buton de trimitere a cererii de actualizare.
* Cazuri limita trate: email-ul introdus nu este asociat niciunui cont, codul de resetare introdus este incorect pentru email-ul asociat, parola nu respecta aceleasi constrangeri amintite la use cas-ul de creare cont.

## 4.4 Autentificare

* Utilizatorul deschide aplicatia
* Este afisata fereastra de login compusa din: campuri pentru email si parola, butoane pentru clear (goleste continutul campurilor), „Change password”, „Not a member? Join now for free”, si buton de „Login”.
* Cazuri limita tratate: email-ul introdus nu este asociat niciunui cont, combinatia email-parola nu este specifica niciunui cont.

## 4.5 Lucrul cu harta autentificat ca “Voluntar”

* Utilizatorul detine un cont de voluntar si se autentifica
* Aplicatia afiseaza un ecran format in partea superioara dintr-un toolbar ce cuprinde: buton de show/hide incidents, buton de adaugare, buton de stergere incident, iconita ce afiseaza poza de profil a voluntarului si buton de logout.

## 4.6 Raportare incident

* Voluntarul apasa butonul „Add incident”
* Apasa apoi pe harta, in locul unde a avut loc incidentul
* Sistemul afiseaza o fereastra noua, cu rol de formular, in care acesta completeaza aspecte privind incidentul si anume: tipul incidentului, gradul de severitate, suprafata afectata, buton de browsing pentru incarcarea unei imagini de la fata locului, text area pentru a adauga descrierea evenimentului si buton de „Save” al incidentului.
* Cazuri limita: voluntarul inchide modala, apasand „X”. Sistemul afiseaza un mesaj de avertizare, care daca primeste un raspuns pozitiv, atunci modala de adaugare incident se inchide si se revine la fereastra principala.

## 4.7 Stergere incident

* Voluntarul apasa butonul „Delete incident”
* Apasa apoi pe harta pe iconita corespunzatoare incidentului ce se doreste a fi sters
* Sistemul cere confirmarea voluntarului privind stergerea
* In cazul in care s-a incercat stergerea unui incident raportat de alt voluntar fata de cel ce este autentificat in sesiunea curenta, se afiseaza un mesaj de eroare si incedentul nu se sterge.

## 4.8 Vizualizare incident

* Voluntarul apasa pe harta pe iconita unuia dintre incidente
* Sistemul afiseaza o modala similara celei de „Adaugare incident”, dar care, de data aceasta vine fara buton de „Save” si este completata cu datele incidentului ales, fara posibilitate de edit a informatiilor.

## 4.9 Lucrul cu harta autentificat ca “Authority”/Entitate legala

* Utilizatorul detine un cont de authority si se autentifica
* Aplicatia afiseaza un ecran format in partea superioara dintr-un toolbar ce cuprinde: buton de show/hide sensors, buton de adaugare, buton de stergere senzor, buton de afisare grafic pentru datele masurate de senzor, luand in calcul ultimele „X” probe, iconita ce afiseaza poza de profil si buton de logout.

## 4.10 Adaugare senzor

* Utilizatorul cu rol de entitate legala apasa butonul „Add sensor”
* Apasa apoi pe harta, in locul unde este amplasat senzorul
* Sistemul afiseaza o fereastra noua, cu rol de formular, in care acesta completeaza valorile initiale ale parametrilor masurati de senzor (turbiditate, ph, salinitate, oxigen, e-coli, enterococcus si sample-rate: intervalul de tip in ore, minute, secunde dupa care senzorul va citi noi valori), si buton de „Save”.
* Cazuri limita: se inchide modala, apasand „X”. Sistemul afiseaza un mesaj de avertizare, care daca primeste un raspuns pozitiv, atunci modala de adaugare senzor se inchide si se revine la fereastra principala.

## 4.11 Stergere sensor

* Entitatea legala apasa butonul „Delete sensor”
* Apasa apoi pe harta pe iconita corespunzatoare senzorului ce se doreste a fi sters
* Sistemul cere confirmarea privind stergerea
* In cazul in care s-a incercat stergerea unui senzor adaugat de alta entitate legala, fata de cea autentificata in sesiunea curenta, se afiseaza un mesaj de eroare si stergerea nu are loc.

## 4.12 Vizualizare date masurate de sensor

* Apasa pe harta pe iconita unuia dintre senzori
* Sistemul afiseaza o modala similara celei de „Adaugare senzor”, dar care, de data aceasta vine fara buton de „Save” si este completata cu datele senzorului ales, fara posibilitate de edit a informatiilor.

## 4.13 Vizualizare grafic cu probele luate de un sensor

* Apasa butonul „View chart”
* Sistemul deschide o modala in care se afiseaza, sub forma de bar chart, datele masurate recent de senzor.

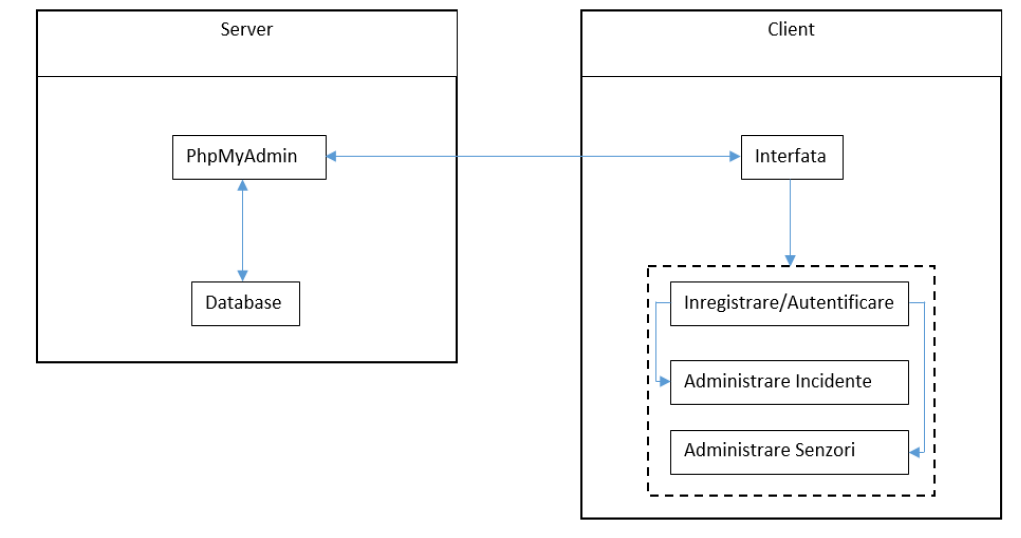
## 4.14 Logout

* Utilizatorul autentificat, apasa butonul de logout.
* Sistemul afiseaza un dialog in care cere confirmarea delogarii

# 5. Arhitectura sistemului

Aplicatia urmeaza modelul software format din doua tipuri de entitati: server si client.

## 5.1 Diagrama arhitecturii de sistem



## 5.2 Serverul

Stocheaza in baza de date informatii legate de conturile utilizatorilor (tip cont, data crearii, date personale, poza de profil), codurile de resetare pentru parole, senzori (coordinate geografice, data amplasarii, rata de obtinere a unor noi date), mostre/probe/samples (sunt masurate valori precum turbiditate, ph, e-coli etc), incidente (tip incident, severitate, coordinate geografice, suprafata afectate, descriere, captura de la fata locului).

Serverul actualizeaza periodic, in functie de rata de obtinere a noi probe specifica fiecarui senzor, valorile afisate de acestia. De asemenea, fiecare set de valori este stocat in baza de date, pentru a putea fi mai tarziu realizate grafice care sa reflecte evolutia in timp a aspectelor monitorizate.

Serverul raspunde cererilor de conectare si creeaza cate o conexiune pentru fecare client. Acest ultim aspect este asigurat de api-ul pus la dispozitie de tool-ul **PhpMyAdmin**, folosit pentru gestiunea bazei de date si interactiunea cu clientii. Serverul raspunde cererilor scrise in limbaj SQL, venite din modulul client, cereri formate de partea de backend in functie de actiunile utilizatorului in frontend.

## 5.3 Clientul

Pentru a putea utiliza programul, este necesar un calculator cu specificatii minime, o conexiune la internet stabila si executabilul in sine. Clientii pot apoi in prima faza sa isi creeze un cont, fie de voluntar, fie de entitate legala/autoritate. Pentru creare, este necesara doar o adresa de email valida si careia sa nu ii fie deja asociat un alt cont.

Un voluntar poate doar sa adauge incidente, sa le stearga pe cele raportate de el, sa vada altele ale altor useri fara a le modifica. O entitate legala are drepturi de create, read pe senzori, poate sterge cei plasati de el si poate vizualiza incidentele raportate de voluntari, fara a le modifica.

Dupa crearea contului, este posibila autentificarea folosind emailul si parola aleasa. Sistemul determina automat tipul contului si afiseaza fereastra corespunzatoare rolului detinut. Se pot realiza apoi actiuni precum: adaugare de senzori, raportare de incidente, stergere (doar pentru entitatile detinute), vizualizare de grafice pentru valorile masurate de senzori. De asemnea, se poate reseta parola asociata contului priind un cod de resetare prin email.

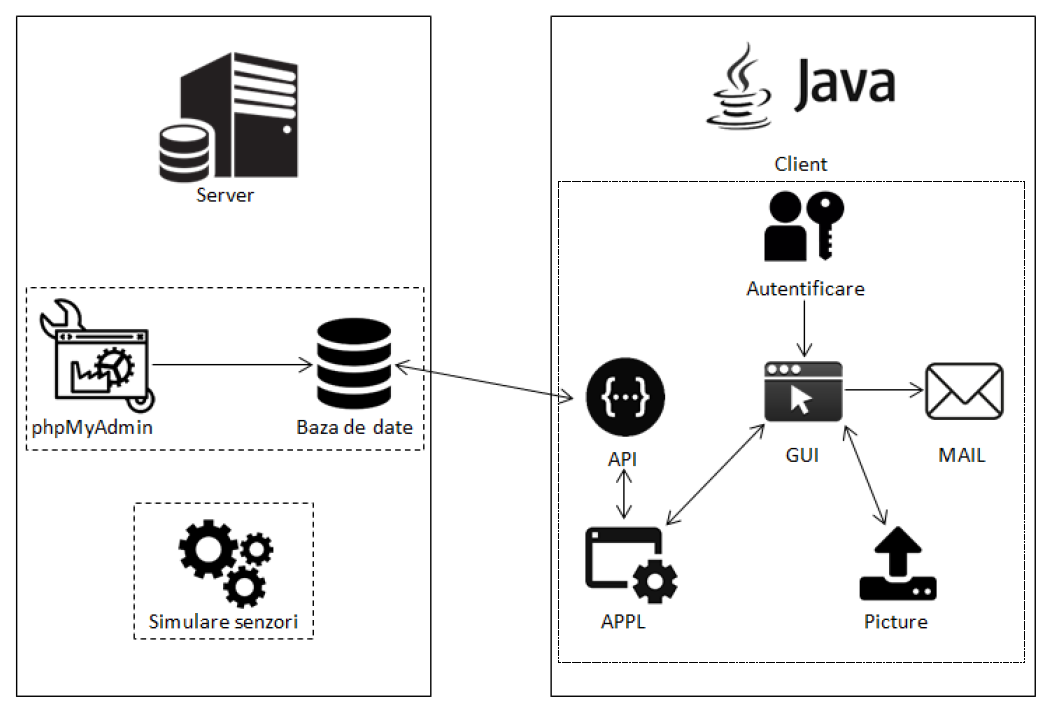
Ideea de functionare a entitatii client este legata de crearea in backend a unor cereri de tip SQL in functie de actiunile realizate in interfata, rulate pe server prin intermediul unei conexiuni (unica in cadrul unei sesiuni) urmand ca serverul sa ofere apoi raspunsul sub forma unor liste sau obiecte definite in aplicatie, din care se extrag apoi informatiile cheie si afisate pe ecran.

# 6. Descrierea componentelor

Aplicaţia constă din următoarele module principale:

* Modul server
* Modul client (aplicatie)

## 6.1 Diagrama de module



## 6.2 Modulul server

Intrucat tool-ul **PhpMyAdmin** pune la dispozitieo baza de date MySql catre care pot fi trimise interogari, urmand a fi prelucrate si oferite raspunsuri, mare parte din functionalitatea pe care trebuia sa o realizeze acest modul este gata implementata. Datoria noastra a fost doar sa obtinem hosting din partea unui server dedicat (johnny.heliohost.org), sa cream structura tabelara a entitatilor in baza de date, sa aplicam constrangeri asupra coloanelor si legaturi intre tabele.

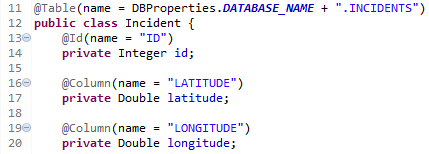
Functionalitatea secundara implementata de noi in acest modul a fost partea de simulare a comportamentului senzorilor.

## 6.3 Modulul client

Este format din mai multe submodule:

### 6.3.1 Modulul api

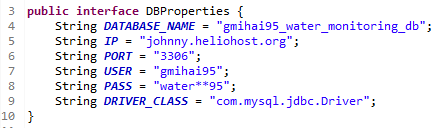
In principal, face maparea/conversia intre o intrare din o tabela de pe server si un obiect Java asociat. In acest scop, am folosit adnotari astfel:



**Explicatie:** Obiectele de tip Incident in Java vor fi mapate pe intrari din tabela INCIDENTS din baza, prin adnotarea @Table. Campul id al obiectului se va mapa pe coloanal “ID” (totodata primary key) din tabela INCIDENTS. Similar, proprietatea latitude va fi asociata coloanei LATITUDE a tabelei.

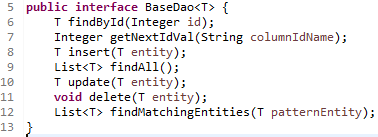
Am folosit apoi foarte mult procedeul numit “reflection”, specific Java pentru a realiza aceste asocieri si pentru a forma cererile de tip SQL si a asigura persistenta datelor.

De asemnea, modulul api asigura conectarea la baza de date. Se ofera o singura conexiune in cadrul unei sesiuni (este implementata folosind Singleton). Credentialele sunt urmatoarele:



### 6.3.2 Modulul appl

Comunica direct cu modulul api (api foloseste reflection si genericitate pentru a crea cereri pentru ORICE tip de obiect Java mapat pe o tabela). Api expune o serie de metode ce pot fi folosite in partea de backend pentru entitatile folosite. Acestea sunt:



### 6.3.3 Modulul Mail

Foloseste protocolul smtp pentru a trimite email-uri din partea de backend. Necesita conexiune la internet, credentialele unui cont valid si are doua mari dezavantaje: functioneaza doar pentru clientii de Gmail si in plus, din contul de Gmail al fiecarui utilizator trebuie bifata optiunea „Allow less secure apps”; Este folosit la obtinerea codurilor de resetare a parolei, si la primirea notificarilor legate de valori ce au depasit limitele normale, masurate de senzori.

### 6.3.4 Modulul Autentificare

Se foloseste un algoritm de criptare a parolelor, astfel incat orice parola introduce utilizatorul, in baza va fi stocata sub forma unui string unic de 128 de caractere. La autentificare, se cripteaza parola si se compara cu cea din baza. De asemenea, tot aici se verifica constrangerile legate de parola: lungime minima, caractere permise in cadrul parolei etc.

### 6.3.5 Modulul GUI

Cuprinde toate ferestrele afisate in aplicatie. Tot aici se afla si logica de trecere de la o fereastra la alta, tratarea evenimentelor de interfata (click, completare campuri, modificare dimensiune fereastra, inchidere fereastra).

In aceasta situatie aplicam design pattern-ul **Observer**. Folosim o entitate principala, echivalentul ferestrei mari, cu rol de Observator, in care doar afisam celelalte view-uri. Celelalte view-uri implementeaza **Subject** si in functie de evenimentele interne, trimit notificari catre fereastra Observator, care va afisa apoi un alt view.

Modulul GUI cuprinde si elemente de interfata “custom” (extensii ale celor cuprinse in pachetul javax.swing).

Printre acestea se numara: fonturi custom, field-uri numerice, text areas cu numar predefinit de randuri si coloane.

Totodata, modulul GUI cuprinde si un submodule pentru lucrul cu imagini.

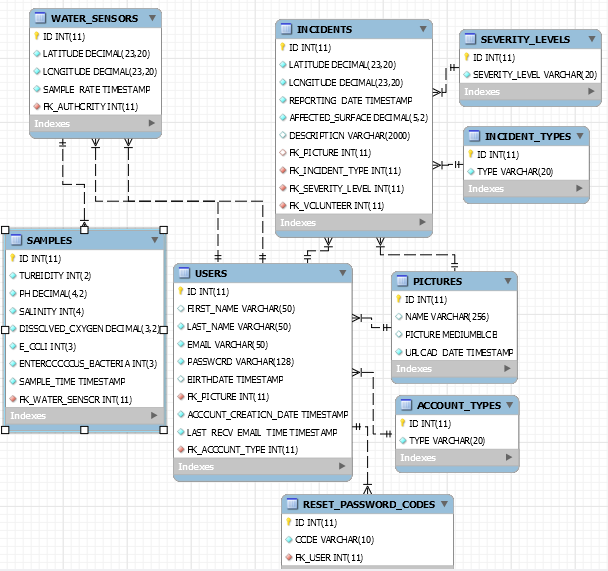
### 6.3.6 Modulul Picture

Are o fereastra pentru browsing, cu ajutorul careia se pot incarca din computer imagini in aplicatie, pentru profilul utilizatorului, captura de la fata locului a incidentelor.

De asemenea, are o componenta ce face descarcarea unui obiect de tip BLOB (Binary Large Object), din baza si conversia lui in imagine pentru a fi afisat utilizatorului. Iconitele aflate pe butoane si alte imagini folosite in ecrane sunt si ele stocate in baza, descarcate, prelucrate de aceasta si expuse.

# 7. Baza de date

## 7.1 Diagrama bazei de date



## 7.2 Tabelele bazei de date

* **SEVERITY\_LEVELS –** are 4 valori posibile: Low, Moderate, Major, Critical
* **INCIDENT\_TYPES** – are 19 valori posibile, de exemplu: Blizzard, Earthquake, Drought, Wildfire etc
* **ACCOUNT\_TYPES** – are 2 valori posibile: Authority si Volunteer
* **RESET\_PASSWORD\_CODES –** asociaza unui user un cod de resetare a parolei, primit prin email. La un moment dat un user are asociat cel mult un astfel de cod, dupa folosirea codului acesta este sters din baza
* **PICTURES –** stocheaza iconitele folosite in aplicatie pentru butoane sa, precum si pozele de la fata locului incidentului sau cele de profil ale utilizatorilor. Un user are o singura poza (sau niciuna), la fel si incidentul
* **WATER\_SENSORS** – un senzor este salvat ca entitate prin coordonatele geografice la care este plasat, rata de actualizare a datelor masurate (reprezentata de ore, minute secunde intre 2 citiri consecutive). Acesta apartine unei singure entitati legale
* **SAMPLES –** sample reprezinta o proba/o masuratoare a unui senzor. Aceasta contine valori pentru turbiditate, ph, salinitate etc, data la care a fost obtinuta precum si legatura catre senzorul care a realizat masuratoarea
* **USERS –** utilizatorii pot fi clasificati in Voluntari si Entitati legale prin intermediul coloanei FK\_ACCOUNT\_TYPE. Pe langa informatiile persoanale de profil, se mai retine despre fiecare user data crearii contului, si data si ora la care a primit ultimul email legat valori in afara normelor pentru senzorii adaugati de el (valabil doar pentru Entitati legale). Acest ultim aspect are rolul de a limita numarul de email-uri pe care o autoritate le poate primi la un singur email pe zi.
* **INCIDENTS** – incidentele sunt raportate de Voluntari, astfel ca pe coloana FK\_VOLUNTEER \_ID se va afla mereu id-ul unui user cu cont de tip voluntar. Un incident are asociat obligatoriu un tip, un grad de severitate si optional, o poza.

# 8. Tehnologii folosite

Pentru partea de server folosim **johnny.heliohost.org** pentru hosting, serviciul **PhpMyAdmin** oferit de ei pentru baza de date si comunicarea dintre server si clienti.

MySql este folosit pentru cererile trimise de la client catre server.

Partea de harta, conversia intre coordonatele geografice si cele spatiale ale hartii expuse in interfata sunt realizate folosind api-ul oferit de ArcGis pentru Java.

Folosim Maven pentru a obtine jar-urile necesare aplicatiei de pe repo-ul oficial maven central. In rest, aplicatia este realizata numai in Java.

Backend-ul foloseste printre altele si facilitati specifice Java8 precum: adnotari si lambda expressions. Interfata este facuta cu ajutorul componentelor din javax.swing si tratarea evenimentelor se face cu ajutorul java.awt.

# 9. Riscuri ce impiedica terminarea aplicatiei

Ca in cadrul dezvoltarii oricarui proiect exista riscuri ce ne pot intarzia terminarea la data stabilita a aplicatiei. In cazul echipei noastre riscurile depistate sunt:

* **Acomodarea cu tehnologiile folosite**

Desi am lucrat in cadrul laboratoarelor atat cu Java cat si cu ArcGIS, cunostintele noastre legate de aceste tehnologii erau minime la inceputul proiectului. Timpul de acomodare si invatare a tehnologiei ArcGis poate ingreuna derularea proiectului.

* **Suprapunerea cu alte teme**

Proiectul, desi anuntat din timp, a inceput intr-o periada foarte aglomerata a semestrului si se suprapune cu aparitia temelor si proiectelor de la alte materii, iar deadline-urile pentru fiecare etapa s-au suprapus cu deadline-urile celelalte.

* **Sincronizarea coechipierilor**

Este dificil sa gasesti intervale orare in care toti membrii sa poata lucra, deoarece programele materiilor optionale nu se poatrivesc, iar pentru o buna desfasurare a proiectului este nevoie de comunicare continua.

# 10. Metode de a minimaliza riscurile

Pentru a minimiza riscul aparut din perspectiva folosirii tehnologiilor necunoscute, putem urmari tutoriale online pentru a ne putea imbunatati cunostintele inaintea inceperii propriu-zise a proiectului.

O buna sincronizare si comunicare in echipa, cu siguranta vor reusi sa remedieze problemele prezentate de ultimele 2 riscuri enuntate.

# 11. Descrierea zonei monitorizate

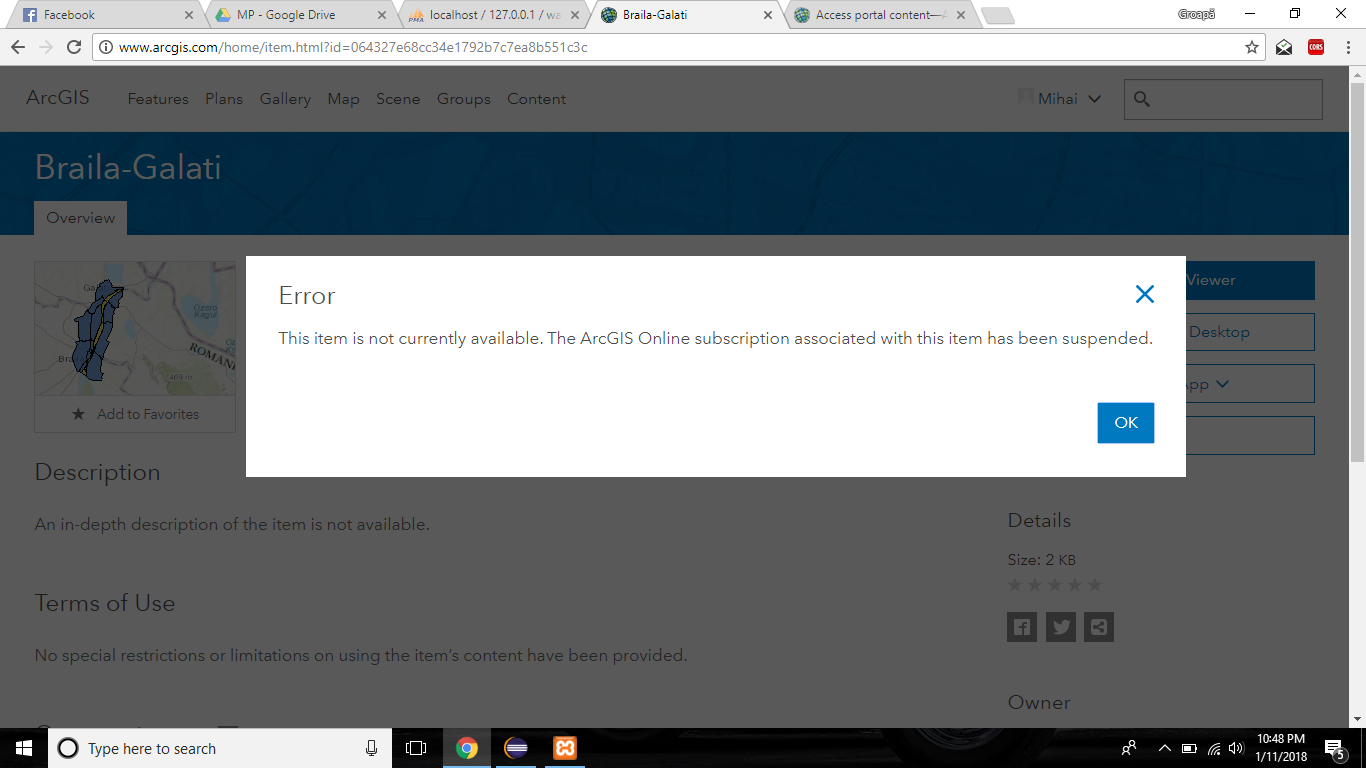
Am ales ca zona de interes portiunea Dunarii cuprinsa intre orasele Braila si Galati. Am realizat layer-ul pentru zona in ArcMap, am incarcat online acest item folosind un cont enterprise. Din nefericire, acesta a avut valabilitate de 20 de zile, astfel incat acum, la predarea proiectului, am avut o supriza neplacuta (contul a expirat cu 2 zile inainte de termen).

In bazinul său superior, între localitățile Braila și Galati, Dunărea se infiltrează complet și dispare, de mai multe ori pe an, pe o lungime de cca 15 km (în medie 155 zile pe an). Apa coboară în subsolul format din straturi de calcar carstificat și reapare la zi în izvorul, situat la cca 14 km la sud de Dunăre.

Din cauza așezării geografice în cadrul continentului a bazinului hidrografic dunărean, la contactul dintre climatul temperat-oceanic din vest, temperat-continental din est și influențele baltice în nord, regimul hidrologic al Dunării se caracterizează prin existența unor importante variații de nivel și de debit în cursul anului și în decursul timpului.

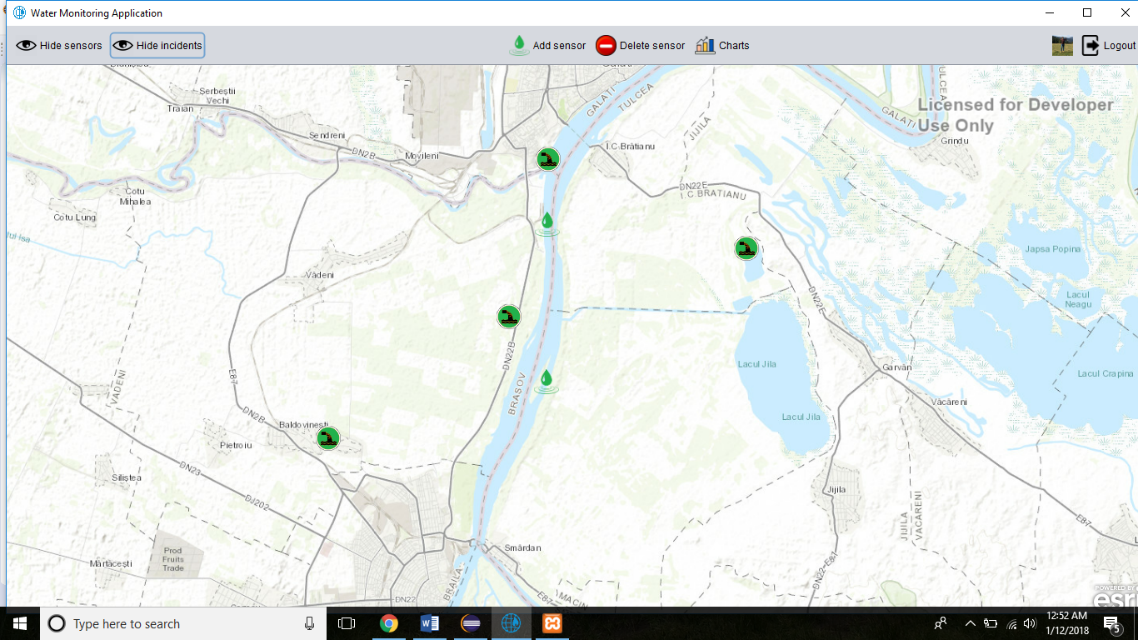
Temperatura apelor Dunării se află sub influența directă a temperaturii aerului și într-o măsură mai mică sub cea a factorilor locali. Încălzirea apelor începe în luna martie și ține până în luna august, după care urmează procesul de răcire. Gheața poate să apară din prima decadă a lunii decembrie până la începutul lunii martie. Durata podului de gheață este în medie de 45-50 de zile. Fenomenul de dezgheț se produce primăvara, cel mai frecvent din aval spre amonte, într-o perioada de câteva zile (4-8 zile).

Dovada:



# 12. Layere

Aplicatia are in prezent 2 layere: unul pentru incidente si unul pentru senzori (acestora li se adauga harta propriu zisa). Pentru rolul de Voluntar, este vizibil doar layer-ul de incidente (acesta nu are dreptul de a accesa datele monitorizate de senzori).



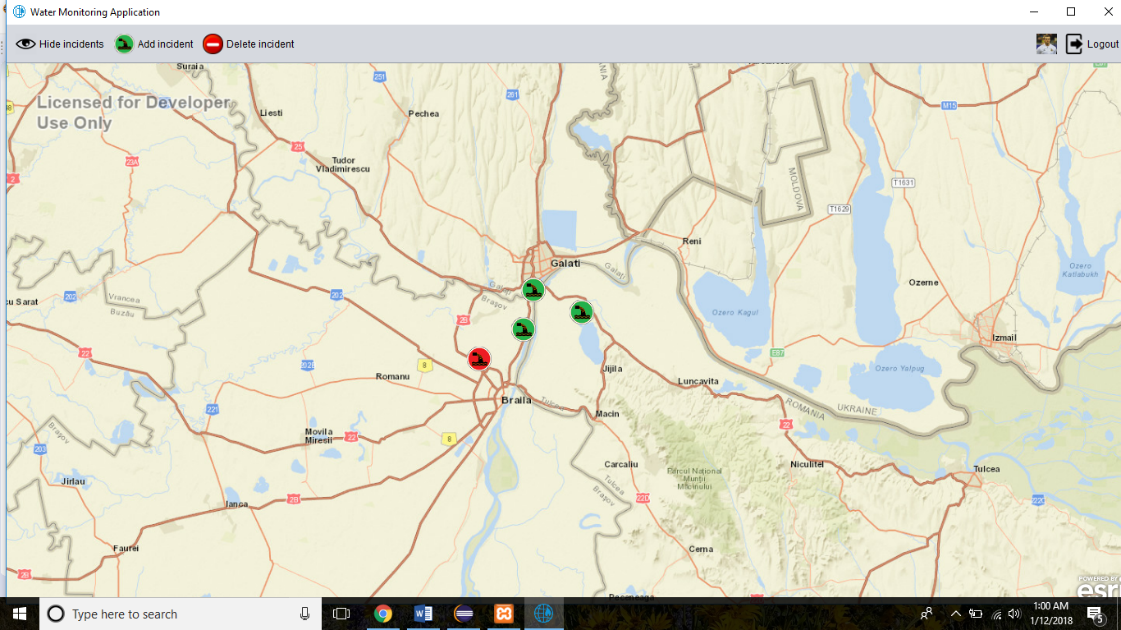
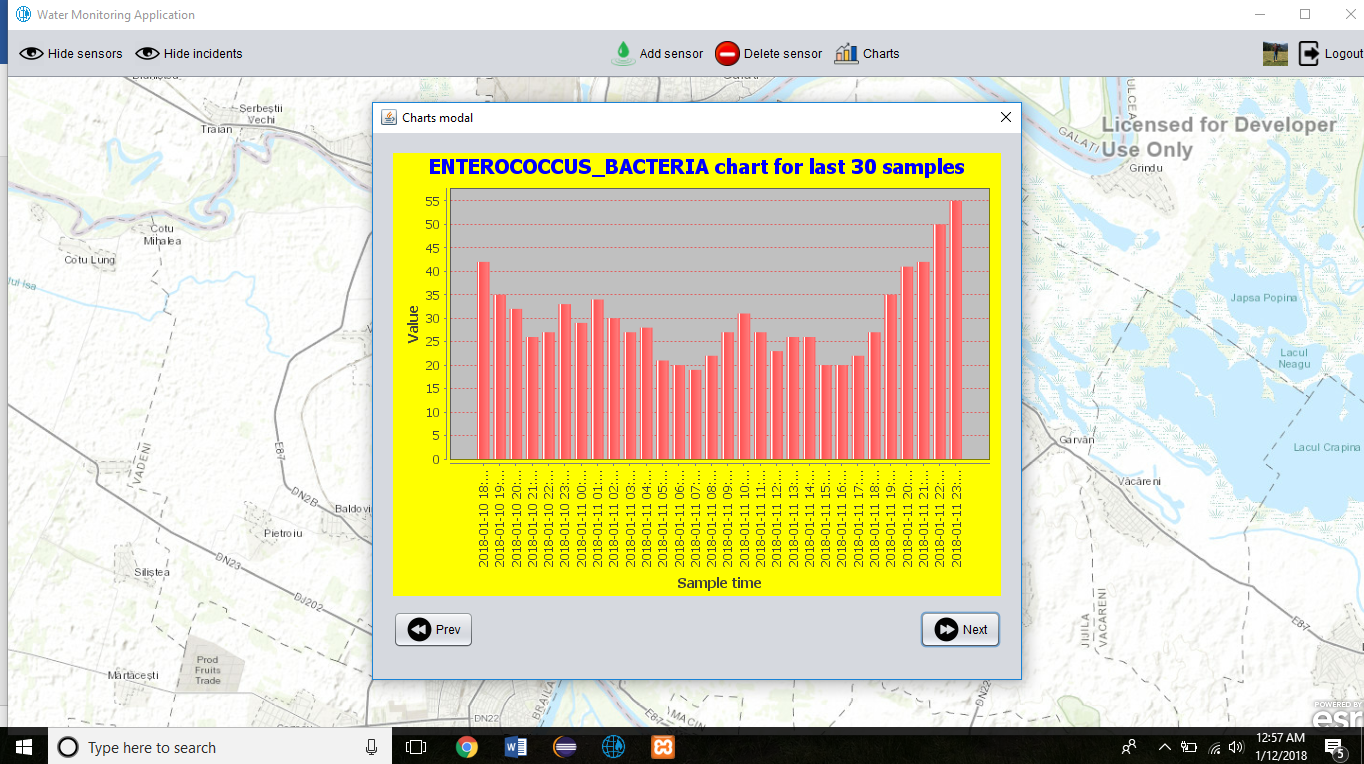


Fig1: ecranul afisat unui user cu cont de tip “authority” (cuprinde ambele layere).

Fig2: ecranul afisat voluntarilor. Incidentele adaugate de alti voluntari sunt marcate prin iconite rosii. Layerul de senzori nu este disponibil.

# 13. Grafice

Graficele afisate sunt de tip barchart. Fiecare sensor are asociat cate un grafic pentru fiecare aspect masurat. Acestea pot fi accesate dand click pe iconita “Charts” din Toolbar, urmand ca apoi sa se dea click pe senzorul dorit. Se afiseaza o modala ce cuprinde graficul current afisat si doua butoane de tip Next/Prev.



De mentionat ca sunt afisate date pentru ultimele 30 de masuratori efectuate de senzor (sau numarul maxim de masuratori, daca nu a efectuat minim 30).

# 14. Referinte

Laboratoarele ArcGis JAVA SDK 1,2 si 3

http://solutions.arcgis.com/state-government/help/water-quality-monitoring/