**Academia de Studii Economice București**

**Facultatea de Cibernetică, Statistică și Informatică Economică**

**Proiect Proiectarea sistemeleor informatice**

**Pîrvu Flavia-Alice**

**Grupa 1054, anul III**

**Informatică Economică**

**București**

**2015**

Cuprins

[1 Prezentarea sistemului informatic 3](#_Toc420548458)

[1.1 Descrierea generală a sistemului informatic 3](#_Toc420548459)

[1.2 Specificarea cerinţelor 5](#_Toc420548460)

[1.2.1 Diagrame detaliate ale cazurilor de utilizare 6](#_Toc420548461)

[1.2.2 Descrierea textuală a cazurilor de utilizare 6](#_Toc420548462)

[2 Analiza sistemului informatic 8](#_Toc420548463)

[2.1 Diagrame de activitate 8](#_Toc420548464)

[2.1.1 Diagrama de activitate publicare date MQTTServer 8](#_Toc420548465)

[2.1.2 Diagrama de activitate pentru modificare valori consumatori 9](#_Toc420548466)

[2.2 Diagrama de clase 10](#_Toc420548467)

[2.3 Diagrame de interacţiune 11](#_Toc420548468)

[2.1.3 Diagrama de secvență pentru modificare valori consumatori 11](#_Toc420548469)

[2.1.4 Diagrama de comunicare pentru modificare valori consumatori 12](#_Toc420548470)

[2.4 Diagrame de stare 13](#_Toc420548471)

[2.5 Rafinarea diagramelor UML 13](#_Toc420548472)

[2.6 Diagrame de procese şi 15](#_Toc420548473)

[2.6.1.Diagrama de procese pentru monitorizarea consumatorilor 15](#_Toc420548474)

[2.6.2.Diagrama de coalborare pentru transmiterea comenzilor catreconsumatori 16](#_Toc420548475)

[3 Proiectarea sistemului informatic 17](#_Toc420548476)

[3.1 Diagrama de clase detaliată 17](#_Toc420548477)

[3.2 Proiectarea bazei de date 17](#_Toc420548478)

[3.3 Proiectarea interfeţelor utilizator 17](#_Toc420548479)

[3.4 Diagrama de componente 17](#_Toc420548480)

[3.5 Diagrama de desfăşurare 18](#_Toc420548481)

[4 Implementarea sistemului informatic 19](#_Toc420548482)

[4.1 Tehnologii utilizate în implementare 19](#_Toc420548483)

[4.2 Prezentarea pe scurt a funcţionalităţii sistemului 19](#_Toc420548484)

# Prezentarea sistemului informatic

## Descrierea generală a sistemului informatic

Sistemele automatizate pentru locuințe întalnesc patru provocări importante, și anume costuri ridicate, inflexibilitate, gestiune la un nivel scăzut și dificultăți în ceea ce privește securitatea. Obiectivele pricipale ale acestei lucrări sunt de a crea și de a implementa un sistem automatizat pentru locuință care să fie ieftin și capabil să controleze și să automatizeze o parte din procese, printr-o interfață web ușor de gestionat care să permită ca sistemul să ruleze continuu și să se mențină în parametrii. De asemenea, soluția propusă vine cu o flexibilitate mare prin folosirea tehnologiei fără fir ce interconectează modulele distribuite la serverul locuinței automatizate. Acest fapt va duce la scăderea costurilor de implementare și va crește abilitatea de îmbunătățire și reconfigurare a sistemului.

Sistemul va folosi sistemul LAN de conexiune la rețea, fără fir pentru a realiza comunicarea dintre modulele hardware distribuite și server, iar comunicarea dintre useri si server se va realiza prin intermediul protocoalelor.

În comparație cu soluțiile existente pe piața fie că sunt “open source”, fie că sunt sisteme dezoltate de către companii, proiectul pe care îl dezvolt încearcă să îmbine managemenul de energie, costuri reduse, dar și o interfață placută și ușor de utilizat.

* Reducerea costurilor de instalare se datorează prin folosirea a cât mai multe componente care pot funcționa fără fir. Pentru componentele care folosesc fire pentru instalare, este nevoie de cabluri și alte materiale adiționale, dar și intervenția unui om specializat în acest domeniu.
* Ușor de implementat, instalat și poate fi folosit pe o arie mai largă. Punctele wireless pot fi montate aproape oriunde, chiar și în locuire greu accesibile pentru componentele conectate cu fire. Tehnologia wireless aduce ca beneficiu lărgirea zonei de acoperire.
* Scalabilitatea sistemului și ușurința de extindere.

Multe dintre soluțiile existente pe piață care deși la prima vedere par flexibile, în realitate constrâng utilizatorii și îi forțează să își adapteze sistemul în funcție de opțiunile pe care le oferă reespectiva companie. Acest fapt, restriționează destul de tare clienții și îi obligă într-un mod tacit să rămâna fideli companiei, așadar orice îmbunătățire a sistemului va depinde de compania care furnizează serviciile de automatizare a locuinței. Restricțiile care apar țin atât de compatibilitatea componentelor sistemului, cât și de o rețea bazată pe conexiunea prin cablu.

Implementarea unei rețele wireless este avantajoasă în primul rând în prisma faptului că ar putea să apară cerințe/specificații noi, sau acestea ar putea să se modifice, iar aceste schimbări vor atrage de la sine extinderea rețelei. În contrast cu instalațiile bazate pe conexiunea prin cablu, nodurile nou adăugate nu necesită cablu în plus, fapt ce ar face extensia trivială.

* Beneficii estetice. Nu doar că plasarea nodurilor este simplă și ca aria de acoperire a sistemului este mult mai mare, dar este vorba și de valențe estetice. Câteva exemple concludente pot fi reprezentate de construcțiile care au în componența arhitecturală doar sticlă, dar și clădirile istorice al căror design sau ale căror motive conservatoare nu permit instalarea unor cabluri adiționale.
* Integrarea dispozitivelor mobile. Asocierea de dispozitive mobbile precum PDA-uri sau smarphone-uri la un sistem automat este posibil oriunde și oricând atâta timp cât nu mai este o problemă prezența efectivă a dispozitivului, existând posibilitatea de control și de la distanță.
* Interfață “user friendly”. Posibiitatea ca utilizatorul să beneficieze de o interfață simplă, prin care să poate coordona un sistem complex se poate realize direct de pe PC sau de pe un dispozitiv mobil, accesând cu ușurință o interfață bazată pe servicii web, local sau la distanță.
* Securitatea și autentificarea.

Doar un utilizator autentificat se poate conecta la sistem – local sau la distanță – pentru a gestiona, controla și a monitoriza. În cazul în care sistemul detectează intruși, acesta trebuie imediat să transmită o notificare către utilizator prin care să-l atenționeze de acest incident și să blocheze posibilitatea de logare pentru o perioadă.

Acest aspect se mai poate adresa și utilizatorilor care nu doresc să folosească chei, ci mai degrabă preferă sa deschidă ușa conectându-se la sistem prin intermediul unui user și a unei parole. Astfel, pot evita problema pierderii cheilor și totodată îi pot ține pe cei mici departe de locurile în care aceștia nu au acces, de exemplu, dulapul cu medicamente, obiecte ascuțite etc.

* Economie de resurse.

Aceste resurse nu se referă doar la cablurile necesare instalării sistemului automatizat. Scopul acestei soluții este să ofere facilități, dar să reducă și costurile, consumul de curent, sau de căldură. Așadar utilizatorul poate avea acces la managementul resurselor și poate decide când să pornească sistemul de încălzire și la ce temperatură, sau când ar trebui stinse/aprinse luminile în funcție de necesitate, de oră sau chiar în funcție de senzorii care pot detecta mișcarea.

Nu doar utilizatorul direct poate beneficia de această economie de resurse, ci și mediul, prin reducerea de consum a energiei electrice, sau a căldurii.

## Specificarea cerinţelor

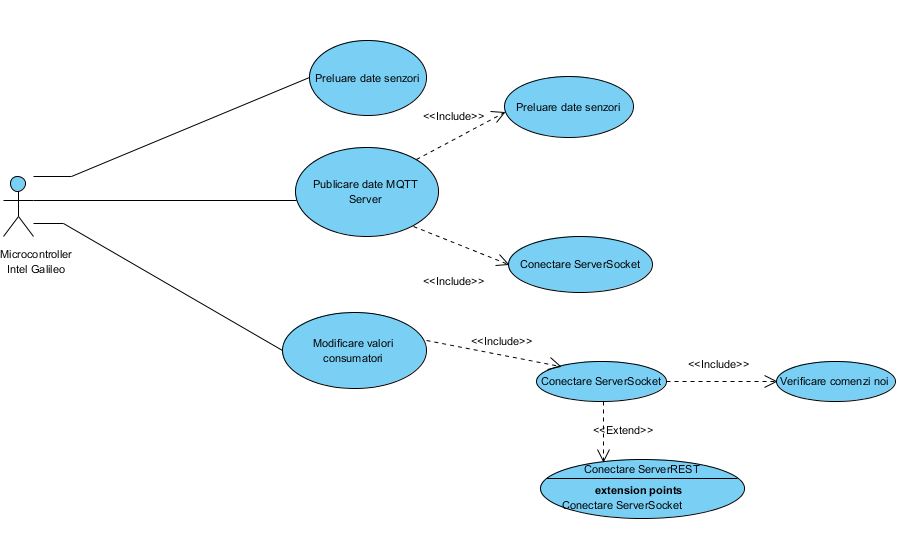
Soluția propusă reprezintă automatizarea unei locuințe sub forma unui sistem distruibuit care cuprinde serverul, microcontrolerul și componentele hardware care reacționează la schimbări. Serverul reprezintă de fapt un PC sau un laptop care se comportă ca un sever web la care se pot conecta utilizatorii, iar acesta coumnică microcontrolerului comenzile primite.

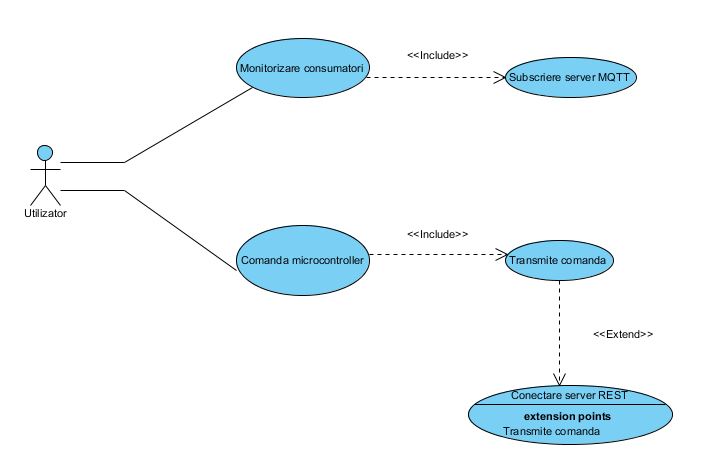
La rândul său, sistemul poate fi accesat prin intermediul unui browser web de pe orice PC local din aceeași rețea LAN folosind IP-ul serverului, sau de la distanță folosind orice dispozitiv care se poate conecta la internet.

Principalele funcții ale serverului constau în a gestioona, a controla și a monitoriza componentele sistemului care acționează în funcție de evenimentele care îi declanșează, sau de comenzile pe care le primesc.

În cadrul acestui capitol vor fi detaliate cerințele funcționale pe care sistemul trebuie să le îndeplinească astfel încât obiectivele anterior menționate să fie atinse.

### Diagrame detaliate ale cazurilor de utilizare



Figură - Diagrama generală a cazurilor de utilizare – perspectivă MICROCONTROLLER

Figură -Diagrama generală a cazurilor de utilizare - perspectivă UTILIZATOR

### Descrierea textuală a cazurilor de utilizare

#### Diagrama cazului de utilizare publicare date MQTTServer

|  |  |
| --- | --- |
| Element al cazului de utilizare | Descriere |
| Cod | CU01 |
| Stare | Schiță |
| Scop | Publicare date citite |
| Nume | Publicarea datelor |
| Actor principal | Microcontroller |
| Descriere | Presupune publicarea datelor în cadrul unui topic specific, date pe care microcontrollerul le preia de la senzorii ce compun sistemul informatic. |
| Precondiții | Microncontrollerul este conectat la rețeaua de internet și are acces la serverul cu care acesta comunică.  Senzorii sunt conectați la microconttroller și sunt funcționali. |
| Postcondiții | - |
| Declanșator | - |
| Fluz de bază | 1. Microcontrollerul este conectat.  2. Preia date de la senzori și încearcă să se conecteze la ServerSocket  3. Verifică dacă există conexiunea cu serverul, dacă da, transmite datele preluate către server, care le va publica la topicul specific. |
| Fluxuri alternative | - |
| Relații | - |
| Frecvența utilizării | Constant |
| Reguli ale afacerii | - |

#### Diagrama cazului de utilizare modificare valori consumatori

|  |  |
| --- | --- |
| Element al cazului de utilizare | Descriere |
| Cod | CU02 |
| Stare | Schiță |
| Scop | Modificarea unor valori |
| Nume | Modificare valori consumatori |
| Actor principal | Microcontroller |
| Descriere | Presupune automatizarea unei locuințe prin modificarea valorilor consumatorilor în funcție de comenzile transmise de la distanță. |
| Precondiții | Microncontrollerul este conectat la rețeaua de internet și are acces la serverul cu care acesta comunică.  Consumatorii sunt conectați la microcontroller și sunt funcționali. |
| Postcondiții | - |
| Declanșator | - |
| Fluz de bază | 1. Microcontrollerul verifică existența unei conexiuni stabilită cu ServerSocket-ul, în caz favorabil așteaptă comenzi de la acesta.  2. La primirea comenzilor, acestea sunt verificate și apoi prelucrate.  3. Se identifică consumatorii.  4. Microcontrollerul trasnimite catre consumatori pentru a ajusta statusul prezent al acestora. |
| Fluxuri alternative | - |
| Relații | - |
| Frecvența utilizării | Din oră în oră |
| Reguli ale afacerii | - |

# 

# Analiza sistemului informatic

## Diagrame de activitate

### Diagrama de activitate publicare date MQTTServer

### AD_Publicare.JPG

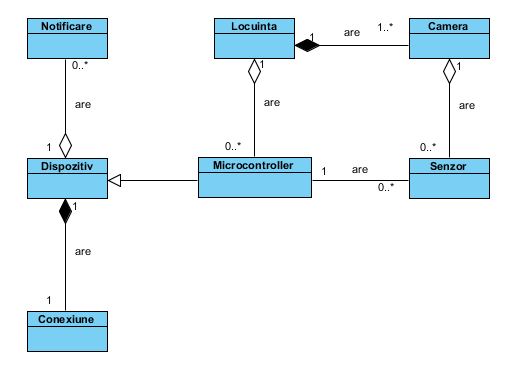
Figură - Diagrama de activitate publicare date MQTTServer

### Diagrama de activitate pentru modificare valori consumatori

### AD_Modificare.JPGAD_Modificare2.JPG

Figură - Diagrama de activitate pentru modificare valori consumatori

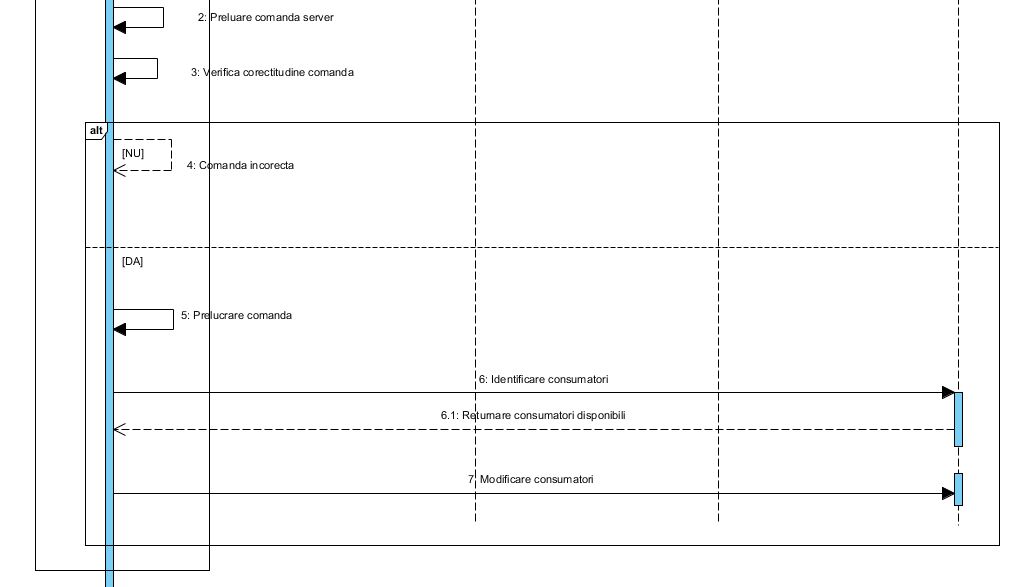
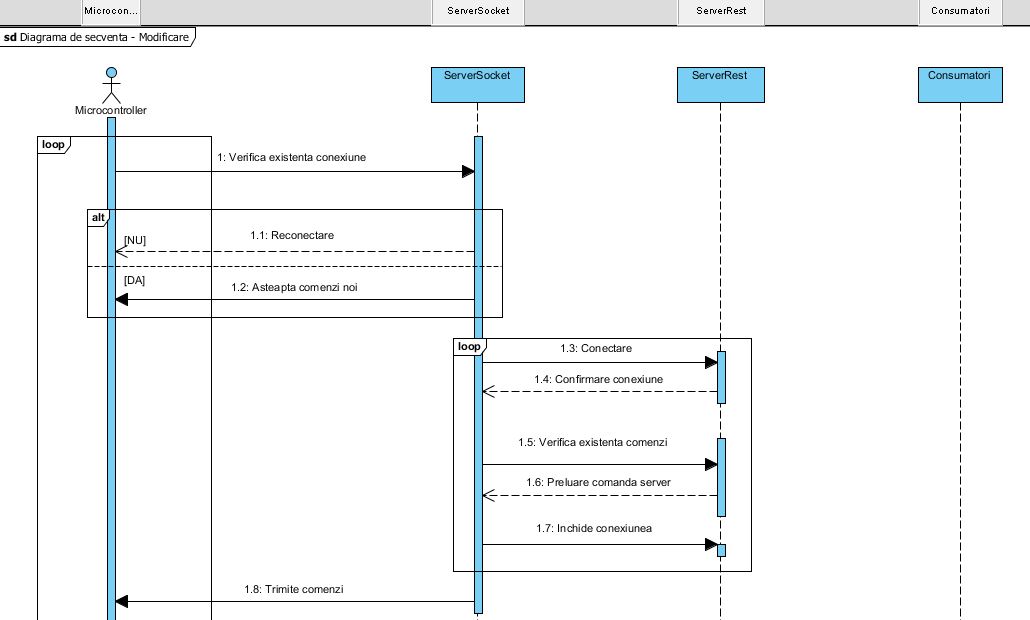
## Diagrama de clase



Figură - Diagrama de clase

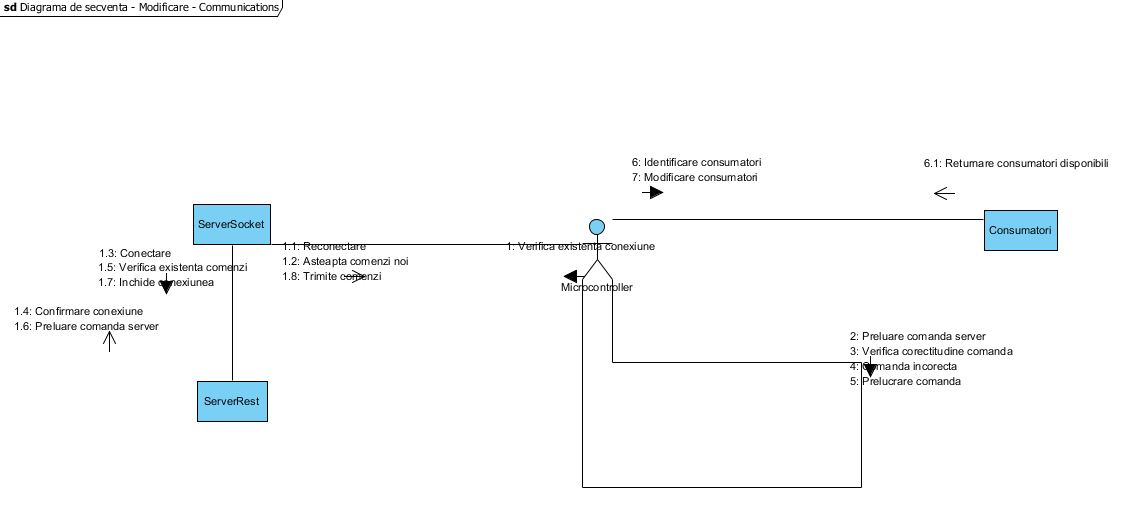
2.3 Diagrame de interacţiune

### Diagrama de secvență pentru modificare valori consumatori



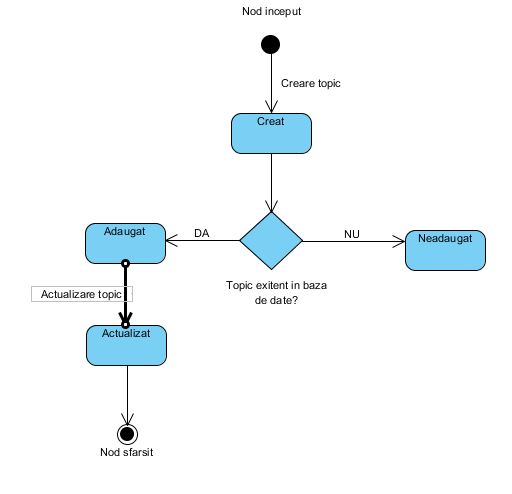
Figură - Diagrama de secvență pentru modificare valori

### Diagrama de comunicare pentru modificare valori consumatori



Figură - Diagrama de comunicare pentru modificare valori consumatori

2.4 Diagrame de stare

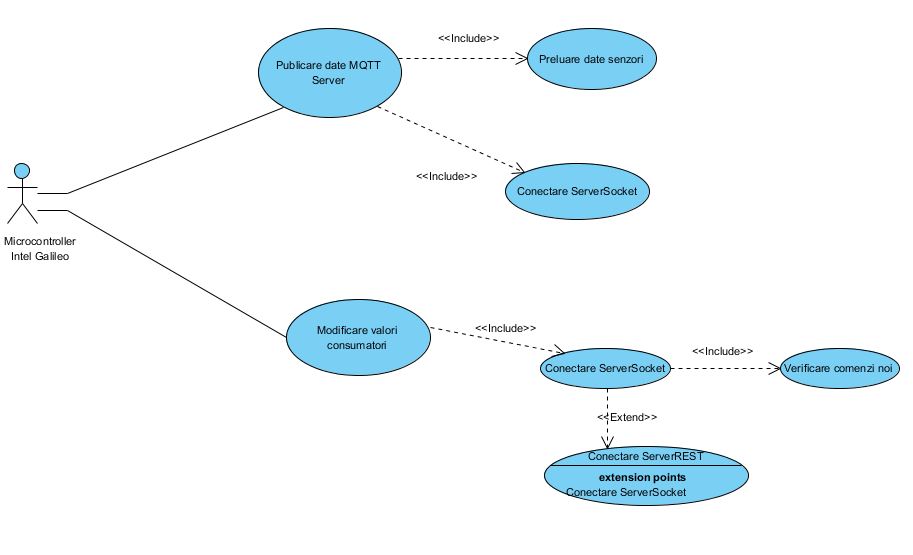


Figură - Diagrama de stare pentru topic

2.5 Rafinarea diagramelor UML

### 2.5.1 Rafinarea diagramei de utilizare - perspectivă microcontroller

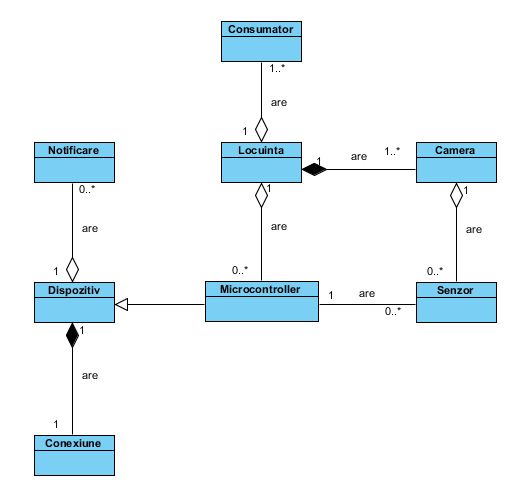
Am ales rafinarea diagramei de utilizare care reprezintă perspectiva microcontroller-ului deoarece am considerat că preluarea datelor de la senzori nu este o utilizare separată a microdontroller-ului ci este un element care relaționează cu cazul de utilizare care publică datele.



Figură - Rafinarea diagramei cazului de utilizare – perspectivă microcontroller

### 2.5.2 Rafinarea diagramei de clase

Am ales rafinarea diagramei de utilizare clase deoarece am constatat lipsea o clasă importantă din cadrul acestei diagrame, și anume clasa Consumator.



Figură - Rafinarea diagramei de clase

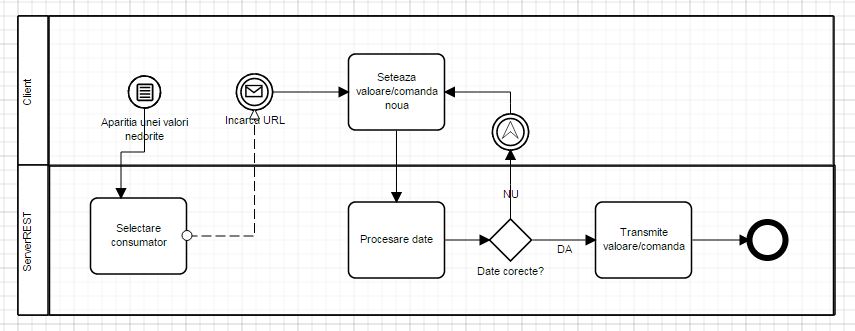
2.6 Diagrame de procese şi colaborare în BPMN

### 2.6.1. Diagrama de procese pentru monitorizarea consumatorilor

### DP - Monitorizare.JPG

Figură - Diagrama de procese pentru monitorizarea consumatorilor

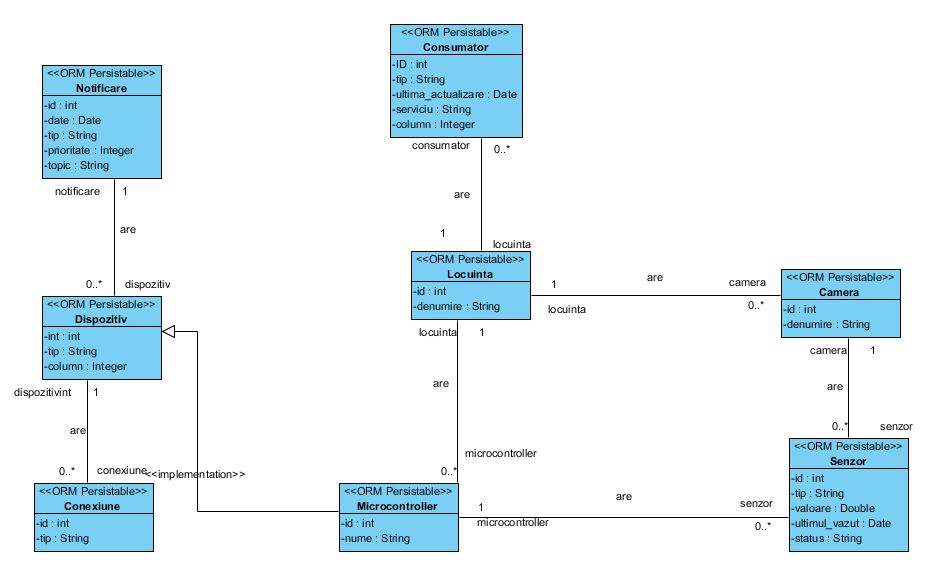
### 2.6.2. Diagrama de coalborare pentru transmiterea comenzilor catreconsumatori



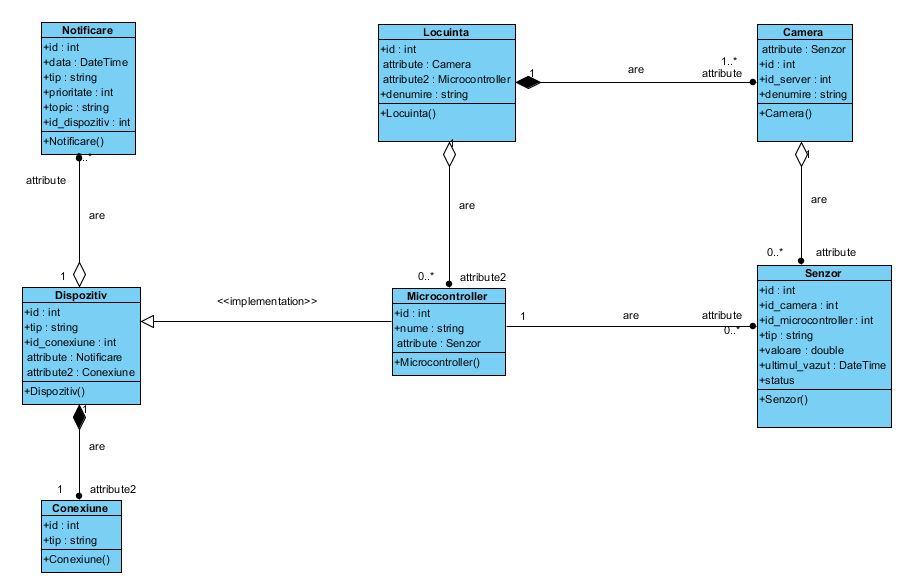
Figură - Diagrama de coalborare pentru transmiterea comenzilor catreconsumatori

# Proiectarea sistemului informatic

3.1 Diagrama de clase detaliată

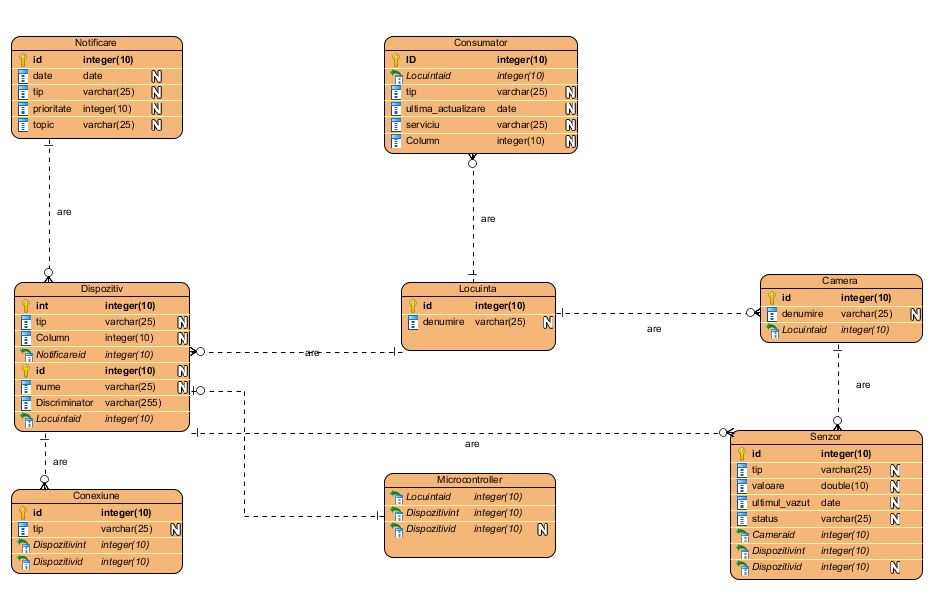


Figură - Diagrama de clase detaliată sincronizată cu schema bazei de date



Figură - Diagrama de clase detaliată

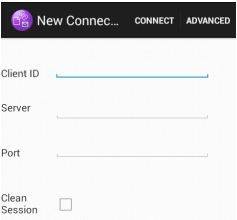
3.2 Proiectarea bazei de date

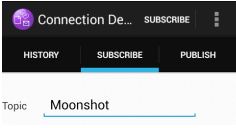


Figură - Schema bazei de date

3.3 Proiectarea interfeţelor utilizator

Modulul Android care are scopul de a se abona la diverse topicuri reprezintă interfața cu utilizatorul intră în contact.

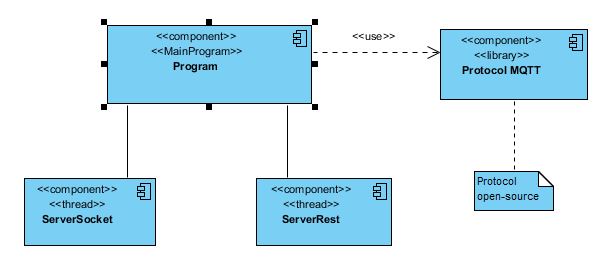




Figură - Interfață utilizator1

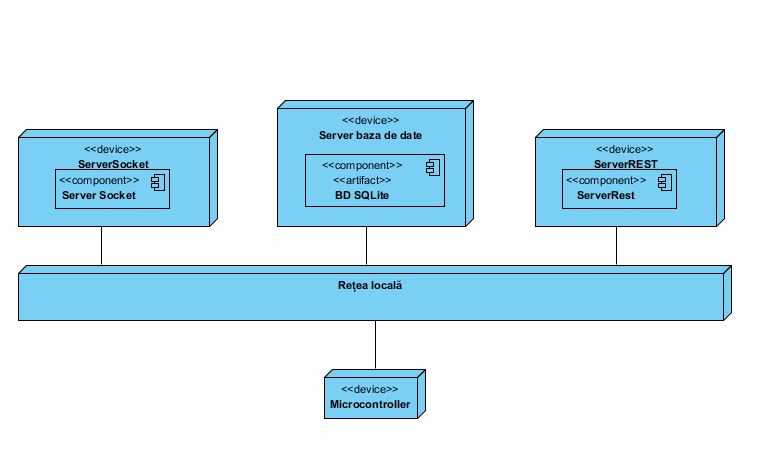
Figură - Interfață utilizator2

3.4 Diagrama de componente



Figură - Diagrama de componente program

3.5 Diagrama de desfăşurare



Figură - Diagrama de desfășurare

# Implementarea sistemului informatic

## Tehnologii utilizate în implementare

Sistemul informatic implementează o serie de tehnologii ce contribuie la buna desfășurare a aplicației de automatizare a unei locuințe.

Pentru microcontrollerul Intel Galileo am folosit platforma open-source furnizată de către Arduino care se bazează pe o placă de microcontroller și pe un mediu de dezvoltare pentru scrierea de software pentru microcontroller. Arduino poate fi folosit pentru a dezvolta obiecte interactive , preluând intrări dintr-o varietate de comutatoare sau senzori și controlează o varietate de lumini , motoare , și alte componente fizice . Proiectele Arduino pot fi de sine stătătoare , sau pot comunica cu software-ul care rulează pe computerul .

Limbajul de programare Arduino este o implementare a electrică , o platformă similară calcul fizic , care se bazează pe procesarea mediului de programare multimedia .

Implementarea celorlaltor componente ale aplicației s-a realizat prin utilizarea limbajului de programare Java și Android în cadrul mediului de dezvoltare Eclipse, folosind facilitățile pe care acesta le oferă. Serverul cu care comunică microcontrollerul este un server pe bazat pe conexiunea prin socket-uri. Serviciile REST sunt susținute de către un server web, open-source dezvoltat de către Apache Software Foundation, și anume, Apache Tomcat. Publicarea datelor în cadrul unui anumit topic, cât și abonarea la un topic se realizează prin intermediul unui protocol de conectivitate destinat comunicări din sfera Internet of Things, acesta fiind implementat atât în limbajul de programare Java cât și Android.

4.2 Prezentarea pe scurt a funcţionalităţii sistemului

Prototipul aplicației are până în prezent o parte din module implementate, printre acestea fiind citirea datelor de la senori și transmiterea acestora către un senzor. Serverul bazat pe conexiunea prin socket-uri este configurat și comunică cu microcontroller-ul, putând să trimită și să primească date. Totodată, Serverul bazat pe socket-uri se conectează la un server MQTT unde publică date. Serviciul REST este configurat și poate primi si transmite unui client date sub formă XML. Abonarea la serverul MQTT este finalizată doar pentru conectarea prin intermediul unei aplicații Java, varainta de aplicație Android este în lucru.