

BAZE DE DATE 2

PROIECT

Aplicație de management a utilităților unui bloc de locuit.

Digori Gheorghe, 343 C5

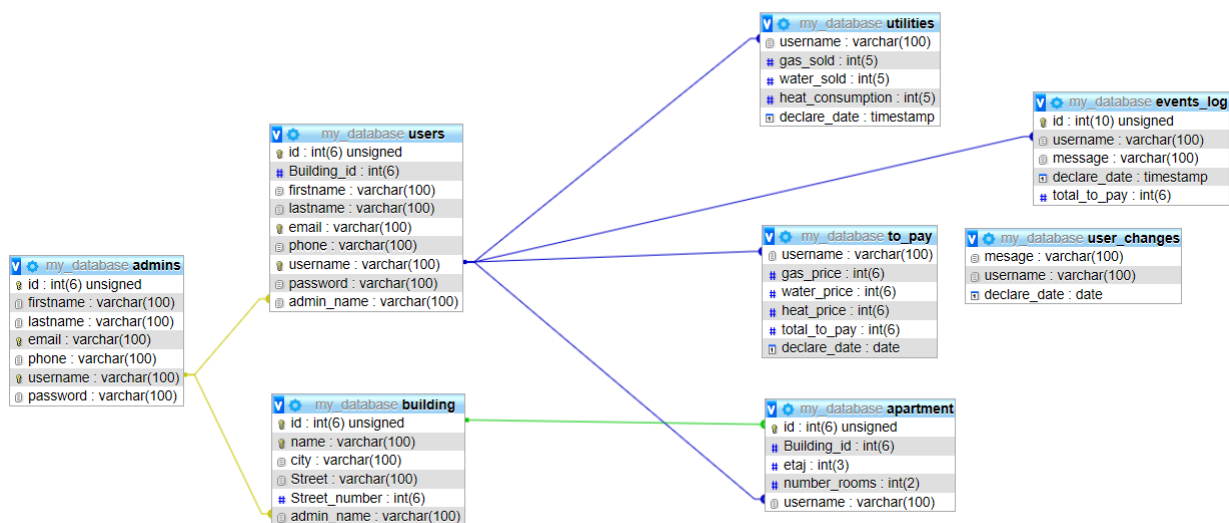
1 .Descrierea temei

Acest proiect a constat in realizarea unei aplicatii de management a unui bloc/ansamblu de blocuri de locuit. In cadrul acestei aplicatii, administratorul se poate loga adauga cladiri noi sau useri noi, Administratorul de asemenea poate vedea activitatea userilor si datele de consum pentru apartamentul acestora. Utilizatorii la randul lor se pot loga cu contul creat de administrator si pot adauga datele de consum curente, pot vedea istoricul de declaratii de date cat si sumele necesare de achitat. Utilizatorii de asemenea au un piechart in meniul de home care reprezinta sumele necesare de achitat pentru utilitatile curente.

2 .Descrierea bazei de date

Sistemul de gestiune al bazei de date este MySQL.

2.1. Diagrama bazei de date



2.2 Structura tabelelor si a constrangerilor de identitate

Denumire	Descriere	Colonane si constrangeri
Users	Retine utilizatorii ce se pot loga in aplicatie	<ul style="list-style-type: none">• ID – indentificator unic cheie primara• Building_id – id-ul cladirii, in care sta utilizatprul, cheie straina la tabela Building• Firstname -nume utilizator• Lastname – familie utilizator• Email - unic• phone• Username - unic• password
Admins	Retine utilizatorii care se pot loga ca administratori	<ul style="list-style-type: none">• ID – indentificator unic cheie primara• Firstname -nume admin• Lastname – familie admin• Email• phone• Username• password

Buildings

Retine lista de blocuri administrate

- ID – indentificator unic cheie primara
- Name – nume cladire Building
- City -Oras(adresa)
- Street – strada adresa
- Street_number – numar bloc

Apartment

Retine lista de apartamente a unui bloc

- ID – indentificator unic cheie primara
- Building_id – id bloc, cheie straina la Buildings
- Etaj
- Number of rooms
- Username – utilizatorului care detine apartamentul, cheie straina la Users

Utilities

Retine datele de consum declarare de un user.

- username – indentificator unic cheie striana la table Users

- gas_sold
- water_sold
- heat_consumption
- declare_date – timestamp

Tabela in care se insereaza preturile pentru utilitati calculare automat la adugarea unei inregistrari in Utilities

- username – indentificator unic cheie striana la table Users

to_pay

- gas_sold
- water_sold
- heat_consumption
- total_to_pay - pret total utilitati
- declare_date – timestamp

	Tabela in care se logheaza datele despre activitatea utilizatorului, adugarea de noi inregistrari la Utilities sau	<ul style="list-style-type: none"> • id – indentificator unic cheie primara
events_log		<ul style="list-style-type: none"> • Username - numele utilizatorului, cheie strina la tablea users • Message – mesajul logat • declare_date - timestamp • total_to_pay – numar total adaugat
user_changes	Tabela de log in care se logheaza activitatea userilor(update, delete) si building-uri	<ul style="list-style-type: none"> • Message – mesajul logat • Username - numele utilizatorului, cheie strina la tablea users • declare_date - timestamp

2.3 .Descrierea procedurilor si functiilor

Tip

Denumire si parametri Descriere

Function

get_userID(username) Returneaza id-ul userului primit ca parametru

Function

return_admin_id(username, pass) Returneaza numarul de id-uri a adminului in functie de username si parola

Function

return_user_id(username, pass) Returneaza numarul de id-uri a userului in functie de username si parola

all_table_rows(tabela) Returneaza toate liniile dintr-o tabela primita ca parametru

Procedure

Procedure

Procedure

Procedure

Procedure

Procedure

Procedure

Procedure

Procedure

Procedure

Delete_declaration(username,
declare_date)

Sterge inregistrarile din Tabela
UTILITIES in functie de username si
data

insert_building(name, city, street,
street_number)

Insereaza o noua inregistrare in
tabela **Building**

insert_user(building_id, firstname,
lastname, email, phone, username,
password)

Insereaza un user in tabela **Users**

insert_utilities(username, gas_sold,
water_sold, heat_sold)

Insereaza o noua inregistrare in
tabela **Utilities**

list_invoices(username)

Returneaza toate liniile din table
to_pay pentru userul primit ca
parametru

list_user_details(username)

Returneaza detaliile userului primit
ca parametru din tabela **Users**

Update_profile(id, username,
building_id, password, firstname,
lastname, phone, email, @Result)

Face update la datele modificate ale
userului in functie de id si returneaza
un paramtru de tip int Result daca sa
produs sau nu vri-o modificare

user_statistics(username)

Returneaza inregistrarile din tabela
Utilities in functie de username-ul
primit ca parametru

utility_price(username, x)

Procedura este apelata de trigerile
delete_invoices si **insert_log** care

Trigger	delete_invoces	determina aduagarea sau stergerea unei inregistrari in tablea Utilities . Procedura primste ca parametru username-ul si un flag(determina trigerul ce apeleaza procedura) si in functie de flagul "x", selecteaza datele introduse in Utilities si calculeaza pretul utilitatiolr si pretul total, daca procedura e apleata de insert_log atunci insereaza rezultatele in tabla to_pay , daca e delete_invoices sterge inregistrarea din table to_pay
Trigger	insert_log	Este declansat la stergerea unei inregistrari din tabla Utilities , apeleaza procedura utility_price pentru a sterge si inregistrarea corespunzatoare din tabela to_pay
Trigger	admin_activity_log_buildings	Este declansat la adaugarea unei inregistrari din tabla Utilities , insereaza un mesaj in tabela events_log si apeleaza procedura utility_price pentru a adauga si inregistrarea corespunzatoare in tabela to_pay
Trigger	admin_activity_log	Insereaza un log in tablea User_changes inainte de inserarea unei linii in tabela building
Trigger	update_profile	Insereaza un log in tablea User_changes inainte de inserarea unei linii in tabela Users
Trigger		Insereaza un log in tablea User_changes dupa modificarea credentialelor unui user din tabela Users

3. Descrierea aplicatiei

3.1 Structura aplicatiei

Partea de frontend a fost implementata folosind html, css, JavaScript. Partea de backend este scrisa in PHP. Aplicatia poate fi pornita prin pornirea unui server cu instanta pe localhost

Aplicatia porneste pe portul 3306 si poate fi accesata la <http://localhost/>.

Interfetele puse la dispozitie sunt, in general, wrappere peste procedurile din baza de date, discutate anterior.

Exista 2 tipuri de useri: Administratori si Utilizatori simpli fiecare dintre acestia avand roluri diferite acces la pagini diferite.

3.2 Workflow

1. Administratorul se logheaza si creaza userii si cladirile.
2. Utilizatorul se logheaza in aplicatie.
3. Utilizatorul este intampinat in pagina de home cu un piechart care reprezinta ce consum current are.
4. Utilizatorul adauga datele de consum current in pagina Declare utilities.
5. Utilizatorul poate naviga in pagina Declared indexes pentru a vedea istoricul datelor declarate.
6. Utilizatorul poate naviga in pagina Invoices pentru a vedea istoricul platilor.
7. Utilizatorul isi poate vizualiza si modifica credentialele accesand pagina Account details.
8. Utilizatorul se poate deloga din aplicatie prin apasarea butonului de Log Out, care il va intoarce la interfata de login.
9. Administratorul poate vizuliza istoricul declaratiilor utilizatorilor accesand pagina Declared utilities.
10. Administratorul poate vizuliza actiunile utilizatorilor asupra profilurilor lor precum si actiunile intreprinse de alti administratori in aplicatie accesand pagina Admin changes.

3.3. Conexiunea cu baza de date

Conexiunea la baza de date se face cu ajutorul pachetului mysqli, prin urmatoarele linii de cod:

```
$conn = new mysqli($servername, $username, $password, $dbname);
```

Dat fiind ca metodele implementate sunt wrapper-e peste functiile si procedurile SQL stocate, se folosesc urmatoarele metode pentru interogarea bazei de date:

proceduri

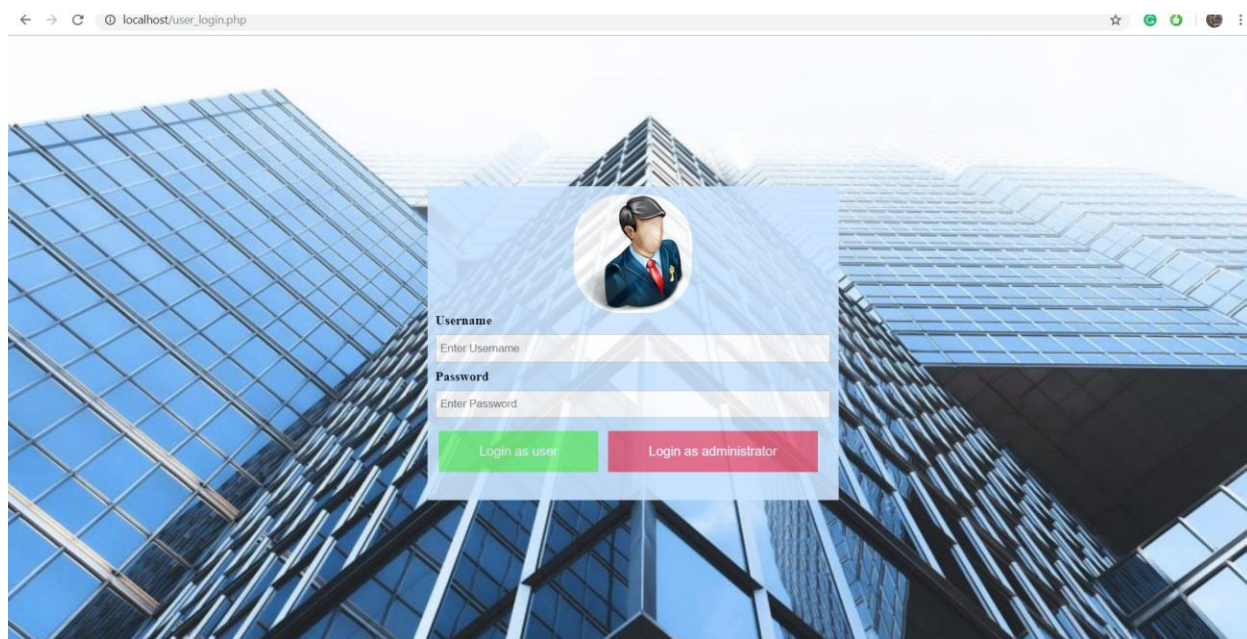
```
$sql = "CALL procedure ('$param')";  
$result = $conn->query($sql);
```

#Functii

```
$sql = "SELECT functie('$param1')";  
$results = mysqli_query($conn,$sql)
```

4. Prezentarea interfetelor si rapoartelor

Utilizatorul (admin sau client simplu) este intampinat de pagina de **Login / Sign Up** unde fiecare tip de utilizator se poate loga confirm rolului sau:



Dupa ce administratorul se logheaza, acesta are acces la 4 interfete: Add, Users Activity, Declared Utilities si Admin Changes.

4.1 Interfete contului de Administrator

4.1.1 Interfata Add

In cadrul acestei interfete adminul poate sa adauge un nou user sau un nou bloc in sistem. De asemenea la click pe sageata de la buton se afiseaza lista de useri sau lista de cladiri.

New block created successfully

Add Users Activity Declared utilities Admin Changes Home Logout

Add user

NEW USER

BuildingID 1

Firstname Alex

Lastname Militaru

Email alex@mail.com

Phone 0761432344

Username alex

Password alex

Add new user

4.1.2 Interfata Admin Changes

La accesarea acestei interfete adminul poate vedea un raport cu evenimentele ce au avut loc pe interfețele admin(ce useri si ce cladiri au fost adaugate) Practic aceasta interfata afiseaza continul tabeli user_changes care logheaza modificarile pe tabela users si building.

Add	Users Activity	Declared utilities	Admin Changes	Home	Logout
Message		name		declare date	
New user has been added		1		2019-01-07	
New building has been added		1		2019-01-07	
New user has been added		gheorghe		2019-01-07	
New building has been added		2		2019-01-07	
New building has been added		Tower		2019-01-07	
New user has been added		alina		2019-01-07	
New building has been added		Sirienilor		2019-01-07	
New user has been added		mihai		2019-01-07	
New user has been added		marin		2019-01-07	
New user has been added		alex		2019-01-07	
New building has been added		Partizanilor		2019-01-07	

4.1.3 Interfata Declared Utilities

Pune la dispozitie raportul cu datele de consum declarate de useeri afisate sub forma unui tabel.

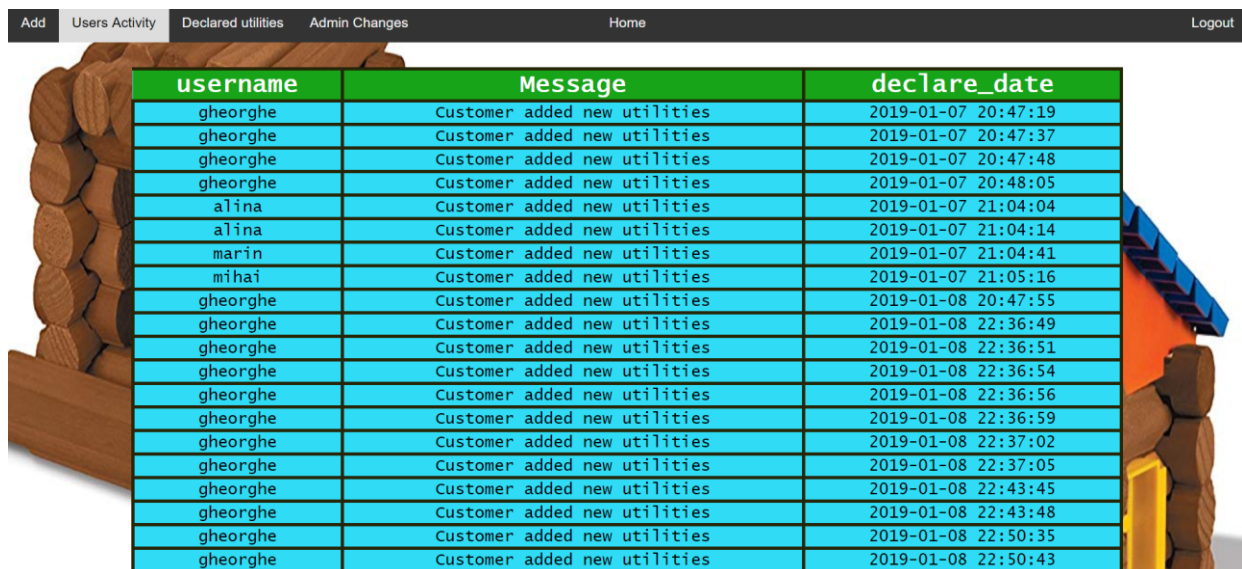


The screenshot shows a web application interface with a navigation bar at the top containing links: Add, Users Activity, Declared utilities, Admin Changes, Home, and Logout. The main content area displays a table with utility declarations. The table has columns for username, gas sold, water sold, heat consumption, declare date, and a delete button. The data is as follows:

username	gas sold	water sold	heat consumption	declare date	delete
alina	2	2	2	2019-01-07 21:04:04	delete
alina	12	14	12	2019-01-07 21:04:14	delete
marin	32	12122	3	2019-01-07 21:04:41	delete
mihai	4	4	4	2019-01-07 21:05:16	delete

4.1.4 Interfata Users Activity

Afiseaza un raport cu loguri fiecare log corespunde unei inregistrari a unui user de date de consum noi.

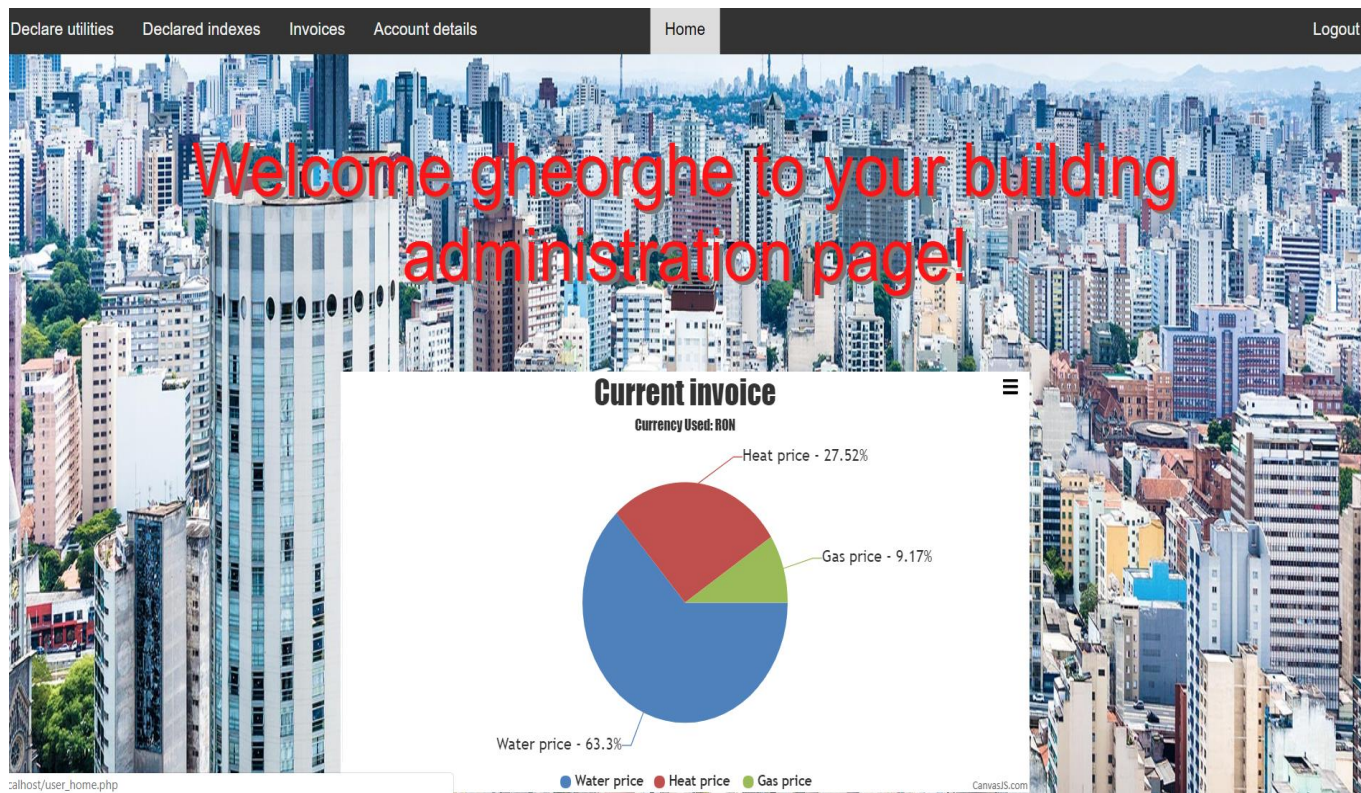


The screenshot shows a web application interface with a navigation bar at the top containing links: Add, Users Activity, Declared utilities, Admin Changes, Home, and Logout. The main content area displays a table with user activity logs. The table has columns for username, Message, and declare_date. The data is as follows:

username	Message	declare_date
gheorghe	Customer added new utilities	2019-01-07 20:47:19
gheorghe	Customer added new utilities	2019-01-07 20:47:37
gheorghe	Customer added new utilities	2019-01-07 20:47:48
gheorghe	Customer added new utilities	2019-01-07 20:48:05
alina	Customer added new utilities	2019-01-07 21:04:04
alina	Customer added new utilities	2019-01-07 21:04:14
marin	Customer added new utilities	2019-01-07 21:04:41
mihai	Customer added new utilities	2019-01-07 21:05:16
gheorghe	Customer added new utilities	2019-01-08 20:47:55
gheorghe	Customer added new utilities	2019-01-08 22:36:49
gheorghe	Customer added new utilities	2019-01-08 22:36:51
gheorghe	Customer added new utilities	2019-01-08 22:36:54
gheorghe	Customer added new utilities	2019-01-08 22:36:56
gheorghe	Customer added new utilities	2019-01-08 22:36:59
gheorghe	Customer added new utilities	2019-01-08 22:37:02
gheorghe	Customer added new utilities	2019-01-08 22:37:05
gheorghe	Customer added new utilities	2019-01-08 22:43:45
gheorghe	Customer added new utilities	2019-01-08 22:43:48
gheorghe	Customer added new utilities	2019-01-08 22:50:35
gheorghe	Customer added new utilities	2019-01-08 22:50:43

4.2 Interfetele si rapoartele utilizatorului de tip Client

Interfata Home a utilizatorului de tip Client intampina userul logat cu un raport sub forma de piechart cu pretul utilitatilor din ultima declaratie.



4.2.1 Declare utilities

In aceasta interfata userul poate declara utilitatile curente completand un formular.

localhost/declare_utilities.php

Declare utilities Declared indexes Invoices Account details Home Logout

Gas Sold*: 10

Water Sold*: 15

Heat consumption*: 100

Submit

4.2.2 Declared indexes

Pune la dispozitia userului raportul cu toate inregistrarile efectuate de user, de asemenea userul poate sterge o inregistrare din raport cu click pe butonul delete.

Declare utilities	Declared indexes	Invoices	Account details	Home	Logout
username	gas sold	water sold	heat consumption	declare date	delete
gheorghe	10	10	100	2019-01-07 20:47:19	delete
gheorghe	5	100	123	2019-01-07 20:47:37	delete
gheorghe	3	33	3	2019-01-07 20:47:48	delete
gheorghe	15	23	10	2019-01-07 20:48:05	delete



4.2.3 Invoices

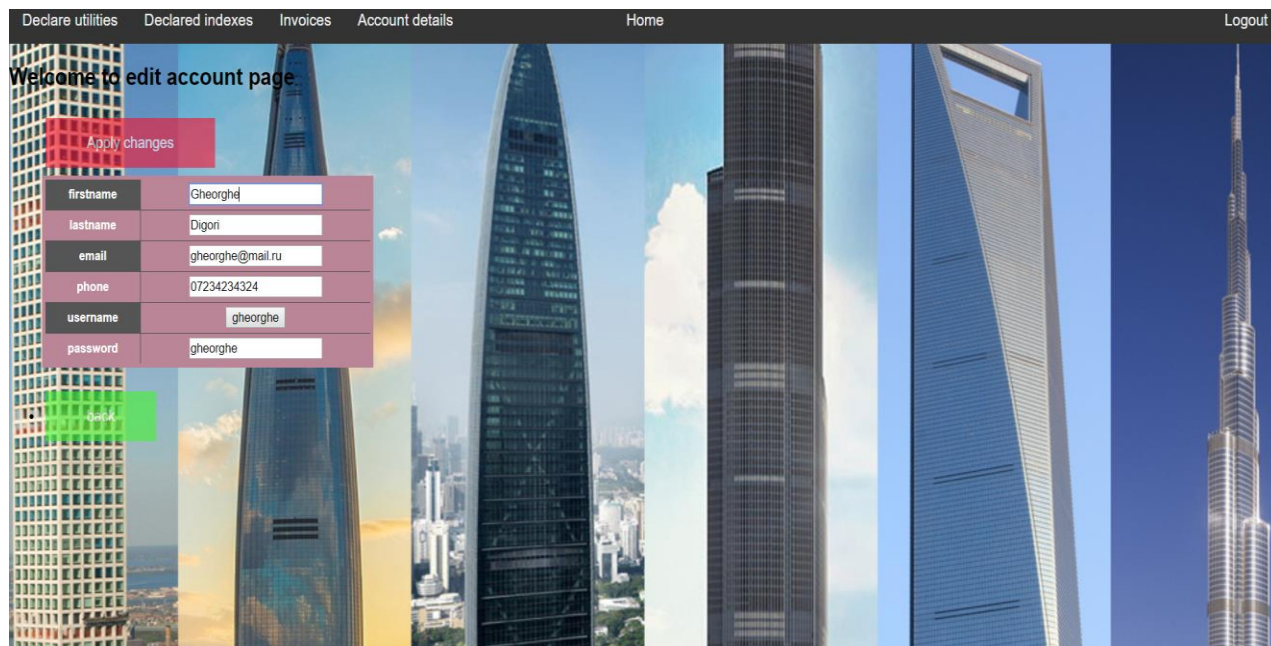
Pune la dispozitia userului raportul cu istoricul preturilor utilitatilor declarate(in parte) precum si pretul total pe care userul curent il are de achitat.

[Declare utilities](#)
[Declared indexes](#)
[Invoices](#)
[Account details](#)
[Home](#)
[Logout](#)

Gas PRICE	Water PRICE	Heat consumption PRICE	TOTAL TO PAY (RON)	LAST declared
100	150	500	750	2019-01-07
50	1500	615	2165	2019-01-07
30	495	15	540	2019-01-07
10	15	5	30	2019-01-08
10	15	5	30	2019-01-08
10	15	5	30	2019-01-08
10	15	5	30	2019-01-08
10	15	5	30	2019-01-08

4.2.4 Account details

Userul logat isi poate vedea datele de profil si poate modifica datele accesand butonul Edit Account userul va fi transferat pe o pagina unde isi poate edita contul si modifica datele, dupa modificare userul va folosi tasta Apply Chnges pentru a salva modificarile.



5. Concluzii

Aplicatia functioneaza conform asteptarilor si cerintelor, toate paghinile sunt active. Operatiile asupra tabelor din baza de date se fac doar prin trigger, proceduri si functii stocate.

Aplicatia are functionalitati de baza si ofera o experienta placuta.

6. Bibliografie

1. <http://programmerblog.net/php-mysqli-tutorial-for-beginners/>
2. <http://www.mysqltutorial.org/mysql-stored-function/>
3. https://www.tutorialspoint.com/android/android_php_mysql.htm
4. <https://dev.mysql.com/doc/refman/8.0/en/create-procedure.html>
5. https://www.w3schools.com/html/html_css.asp
6. <http://sketchytech.blogspot.com/2012/07/starting-out-with-mamp-mysql-and-php.html>