

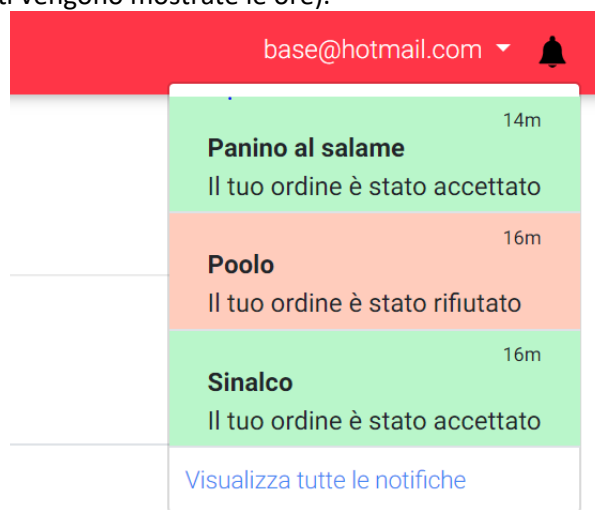
# Diario di lavoro

Luogo	Trevano
Data	17.02.2020

## Lavori svolti

Ho continuato la parte delle notifiche, utilizzando ajax ho creato vado a richiedere le ultime tre notifiche ad un metodo del controller notification.php che ritorna il tutto in formato json.

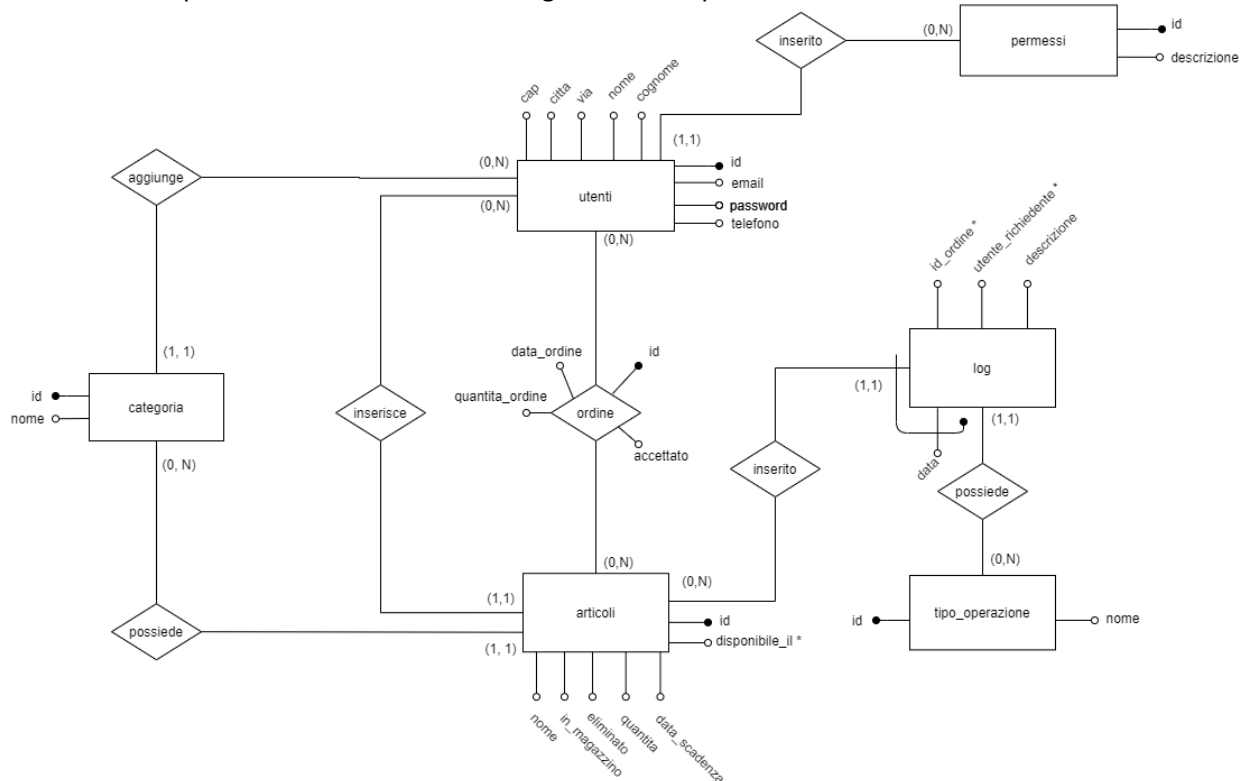
Una volta ricevuto il json aggiungo dei div contenenti le informazioni delle notifiche e, dopo aver calcolato il tempo, viene visualizzato a schermo la data della notifica (dopo le 23 ore viene mostrata la data e l'ora, dopo 59 minuti vengono mostrate le ore).



Premendo sul link “Visualizza tutte le notifiche” si viene indirizzati ad una pagina in cui sono mostrate tutte le proprie notifiche.

Durante lo sviluppo della pagina di notifiche mi sono accorto che dalla tabella di log non ho nessun riferimento agli ordini, quindi non posso recuperare le informazioni di quest'ultimi. Ho dovuto effettuare molte modifiche siccome nella tabella **ordine** non avevo un id ma utilizzavo 3 campi come primary key.

Per evitare di utilizzare 3 campi anche nelle altre tabelle come chiave primaria ho dovuto aggiunger un id e modificare le parti di codice dove utilizzavo gli altri 3 campi.



Questa parte di modifica del codice mi ha fatto perdere un bel po' di tempo.

Ho deciso di mostrare tutte le notifiche dell'utente nell'apposita pagina. Ho utilizzato un modal per visualizzare i dettagli dell'ordine che è stato rifiutato/accettato.

Magazzino merci
Catalogo
base@hotmail.com

## Notifiche

Qui puoi visualizzare tutte le tue notifiche

Mostra 10 record per pagina Cerca:

Articolo	Quantità	Descrizione	Data elaborazione dell'ordine	Dettagli
Pizza	1	Richiesta d'ordine accettata	17.02.2020 10:00	<a href="#">i</a>
Poolo	1	Richiesta d'ordine rifiutata	17.02.2020 10:00	<a href="#">i</a>
Coca cola	3	Richiesta d'ordine rifiutata	17.02.2020 9:59	<a href="#">i</a>
Poolo	3	Richiesta d'ordine accettata	17.02.2020 9:59	<a href="#">i</a>
Pizza	3	Richiesta d'ordine rifiutata	17.02.2020 9:59	<a href="#">i</a>
Panino al salame	1	Richiesta d'ordine accettata	17.02.2020 9:59	<a href="#">i</a>
Poolo	4	Richiesta d'ordine rifiutata	17.02.2020 9:51	<a href="#">i</a>
Poolo	3	Richiesta d'ordine accettata	17.02.2020 9:51	<a href="#">i</a>

Dettagli ordine

×

Questo ordine è stato accettato

ID ordine	20
Articolo	Pizza
Quantità	1
Data scadenza	29.02.2020
Data ordine	17.02.2020 10:00:03

CHIUDI

Una volta fatto ciò mi sono dedicato al report del magazzino (pagina in cui è possibile monitorare la situazione del magazzino). Ho creato una pagina simile alle altre in cui vengono mostrate tutti i record preseti nella tabella di log.

Successivamente ho creato una stored procedura nella banca dati che controlla, tramite un evento, una volta al giorno se stanno per scadere articoli. L'idea è che se l'articolo è scaduto viene settato il campo **eliminato** della tabella articoli a 1, viene inserito nella tabella di log e tutti gli ordini per quell'articolo vengono rifiutati. Inoltre gli articoli che stanno per scadere vengono aggiunti alla tabella di log in modo da poter essere visualizzati dagli admin e gli operatori.

Ho utilizzato il campo facoltativo **utente\_richiedente** della tabella **log** per controllare se è un campo che può essere eliminato o meno. Se non è settato significa che il record non è riferito ad una richiesta d'ordine e nemmeno ad un articolo che è stato eliminato. Così ogni giorno i record che non hanno questo campo settato vengono eliminati automaticamente. Se un articolo è scaduto non viene eliminato il record dal file di log in modo da poter sempre visualizzare gli articoli scaduti.

Procedura che elimina i record degli articoli che stanno per scadere:

```
DELIMITER //
```

```
CREATE PROCEDURE deleteOldRecords()
BEGIN
    DELETE FROM LOG WHERE utente_richiedente IS NULL AND id_tipo_operazione != 4;
END //
```

```
DELIMITER ;
```

Procedura che controlla ed inserisci gli articoli scaduti/in scadenza:

```
DELIMITER //
```

```
CREATE PROCEDURE checkExpiredArticles()
BEGIN
    DECLARE finished int DEFAULT 0;
    DECLARE article_id int;
    DECLARE expire_date date;

    -- declare cursor article
    DECLARE curArticle
        CURSOR FOR
            SELECT id, data_scadenza FROM articoli WHERE eliminato = 0 AND
in_magazzino = 1;

    -- declare NOT FOUND handler
```

```
DECLARE CONTINUE HANDLER
    FOR NOT FOUND SET finished = 1;

-- delete old record
call deleteOldRecords();

OPEN curArticle;

getArticleId: LOOP
    FETCH curArticle INTO article_id, expire_date;
    IF finished = 1 THEN
        LEAVE getArticleId;

    -- article expired
    ELSEIF expire_date <= date(now()) THEN

        -- insert into log
        INSERT INTO LOG(id_articolo, id_tipo_operazione) VALUES (article_id, 4);

        -- delete article from storage
        UPDATE articoli SET eliminato = 1 WHERE id = article_id;

    -- article almost expired
    ELSEIF expire_date <= date(now() + INTERVAL 2 day) THEN

        -- insert into log
        INSERT INTO LOG(id_articolo, id_tipo_operazione) VALUES (article_id, 3);
    END IF;
END LOOP getArticleId;
CLOSE curArticle;

END //
```

DELIMITER ;

Evento che viene chiamato una volta al giorno:

```
/****** Automatic event *****/
SET GLOBAL event_scheduler = ON;

CREATE EVENT daily_articles_check
ON SCHEDULE EVERY 1 day
DO
call checkExpiredArticles();
```

Link sintassi stored procedure:

[https://www.quackit.com/mysql/tutorial/mysql\\_stored\\_procedures.cfm](https://www.quackit.com/mysql/tutorial/mysql_stored_procedures.cfm)

Link cursori: <https://www.mysqltutorial.org/mysql-cursor/>

Una volta fatta questa parte ho mostrato a schermo, nella pagina di report, la situazione del magazzino:

Magazzino merci
Utenti
Categorie
Catalogo
Report
Ordini
admin@hotmail.com

## Situazione magazzino

**Attenzione:** se un articolo scade tutte le richieste d'ordine verranno rifiutate automaticamente

Mostra  record per pagina Cerca:

Articolo	Quantità	Descrizione	Data elaborazione dell'ordine	Dettagli
Quasi scaduto		Articolo scaduto	17.02.2020 14:52	<a href="#">i</a>
Quasi scaduto		Articolo scaduto	17.02.2020 15:00	<a href="#">i</a>
Sinalco		Articolo in scadenza	17.02.2020 15:00	<a href="#">i</a>
Sinalco		Articolo in scadenza	17.02.2020 15:00	<a href="#">i</a>

### Problemi riscontrati e soluzioni adottate

-

### Punto della situazione rispetto alla pianificazione

In anticipo rispetto la pianificazione.

### Programma di massima per la prossima giornata di lavoro

La prossima lezione devo sviluppare la pagina in cui ogni utente base (cliente) può visualizzare la cronologia delle sue richieste d'ordine.