

Diario di lavoro

Luogo	SAMT
Data	03.04.2019

Lavori svolti

Questa mattina l'ho usata quasi interamente a completare la modifica dei dati nel carrello, ovvero aggiungere o togliere prodotti direttamente da lì.

Per farlo ho dovuto modificare diverse classi, tra cui anche quelle modificate ieri, e altre più vecchie, per poter implementare il funzionamento che mi serve.

La prima modifica che ho fatto è stata alla classe già cambiata ieri.

La modifica che ho applicato sta nel dividere lo switch che controlla l'azione che l'utente vuole eseguire, come mostrano le immagini sotto.

```

56      switch ($action) {
57          case "add":
58              $quantity = $this->addQuantity($buyCount);
59              break;
60          case "delOne":
61              $quantity = $this->removeQuantity($buyCount);
62              break;
63      }

```

Figura 1 modifica quantità richiesta

```

97      switch ($action) {
98          case "add":
99
100              //Imposto la nuova quantità del prodotto
101              $product->setQuantity($prodName, $prodPrice, $prodQuantity, $prodQuantity - 1);
102              $prodQuantity--;
103              break;
104          case "delOne":
105
106              //Imposto la nuova quantità del prodotto
107              $product->setQuantity($prodName, $prodPrice, $prodQuantity, $prodQuantity + 1);
108              $prodQuantity++;
109              break;
110          default:
111
112              //Imposto la nuova quantità del prodotto
113              $newQuantity = isset($buyCount[0]['quantita_richiesta'])? $prodQuantity + $buyCount[0]['quantita_richiesta']
114                          : $prodQuantity + $buyCount['quantita_richiesta'];
115              $product->setQuantity($prodName, $prodPrice, $prodQuantity, $newQuantity);
116      }

```

Figura 2 modifica quantità prodotto

La modifica è stata applicata perché la modifica della quantità richiesta va fatta prima, ma quella del prodotto non può essere modificata prima, altrimenti la query viene fatta sul dato sbagliato, quindi le righe mostrate nella seconda immagine vanno inserite appena prima della riga "\$sql->execute();" in modo da far applicare tutto nel modo corretto.

Un'altra modifica che bisogna attuare per far funzionare il tutto è stata all'interno della funzione "getProduct" nella classe model "ProductModel" come mostrato sotto.

```

53 if(count($buy->getDataByProduct($name, $price, $quantity)) > 0) {
54     //Prendo i dati dell'utente in base alla mail
55     $sql = $conn->prepare("SELECT * from prodotto p
56         right outer join compra c on p.nome = c.nome_prodotto and p.prezzo = c.prezzo_prodotto and p.quantita = c.quantita_prodo
57         left outer join vende v on v.nome_prodotto = p.nome
58         left outer join negozio n on n.nome = v.nome_negozio and n.indirizzo = v.indirizzo_negozio and n.citta = v.citta_negozio
59         WHERE p.nome LIKE :nome AND p.prezzo LIKE :prezzo AND p.quantita LIKE :quantita");
60     $sql->bindParam(':nome', $name, PDO::PARAM_STR);
61     $sql->bindParam(':prezzo', $price, PDO::PARAM_INT);
62     $sql->bindParam(':quantita', $quantity, PDO::PARAM_STR);
63 } else {
64     //Prendo i dati dell'utente in base alla mail
65     $sql = $conn->prepare("SELECT * from prodotto p
66         left outer join vende v on v.nome_prodotto = p.nome
67         left outer join negozio n on n.nome = v.nome_negozio and n.indirizzo = v.indirizzo_negozio and n.citta = v.citta_negozio
68         WHERE p.nome LIKE :nome AND p.prezzo LIKE :prezzo AND p.quantita LIKE :quantita");
69     $sql->bindParam(':nome', $name, PDO::PARAM_STR);
70     $sql->bindParam(':prezzo', $price, PDO::PARAM_INT);
71     $sql->bindParam(':quantita', $quantity, PDO::PARAM_STR);
72 }

```

Figura 3 *getProduct controllo prodotto richiesto*

Nella modifica è stato aggiunto un controllo per assicurarsi se c'è o meno un prodotto nella tabella "compra", questo è stato fatto perché la query inizialmente con le join prendeva anche i valori dalla tabella ponte, ma se non i sono dati non ritorna niente, ed essendo questa funzione utilizzata per altre cose oltre che per il carrello deve prendere i dati anche se non ci sono.

Problemi riscontrati e soluzioni adottate

Tutti i problemi che ho avuto sono legati alla modifica della quantità di prodotti nel carrello.

Punto della situazione rispetto alla pianificazione

Oggi ho completato la parte del carrello, ma senza l'eliminazione completa del prodotto, questo perché così non perdo altro tempo in quello ma mi porto avanti.
Devo ancora gestire il caso in cui 2 negozi facciano riferimento allo stesso prodotto e in uno dei 2 diminuisca la quantità.

Programma di massima per la prossima giornata di lavoro

Implementare il checkout.