Diario di lavoro

|  |  |
| --- | --- |
| Luogo | SAMT |
| Data | 18.03.2019 |

|  |
| --- |
| Lavori svolti |
| La prima cosa che ho fatto è stata completare la mostra delle informazioni del carrello appena l’utente accede o quando inserisce un nuovo prodotto nel carrello.  Per fare questo, in JavaScript ho creato delle funzioni che se ne occupano.    Figura 1 Funzione getCart  La funzione che prende tutti i dati del carrello ha subito delle modifiche, non va più a codificare i dati ma passa direttamente i dati alla funzione che li inserisce nella pagina, questo perché la funzione “getCart” del controller passa direttamente i dati dei prodotti e non più quelli del carrello, come mostrato nell’immagine qui sotto.    Figura 2 funzione controller getCart  La funzione prende i dati dal carrello e dopo con quelli prende tutti i prodotti che l’utente ha.  Oltre a questo anche quando viene aggiunto un prodotto al carrello viene richiamata un’altra funzione.    Figura 3 funzione addToCart  La funzione controlla che il prodotto abbia un valora maggiore di 0 e in quel caso va a richiamare la funzione e prende i valori del carrello con quel prodotto, una volta fatto diminuisce il valore che è scritto nella pagina sul prodotto e dopodiché richiama la funzione che prende il prodotto e lo inserisce nel carrello.    Figura 4 funzione getCartProduct JS  La funzione si occupa di prendere il prodotto in base ai dati passati e passarlo alla funzione che li inserisce nella pagina.  Oltre alla modifica della funzione “getCart”, nella classe controller “Customer” ci sono altre funzioni nuove, la funzione che inserisce i dati nella tabella “addToCart” e quella che prende il prodotto singolo da aggiungere al carrello.    Figura 5 funzione addToCart  La funzione prende i valori della chiave del prodotto e poi inserisce le informazioni nella tabella grazie alla funzione model “insertData”, che ha anch’essa subito dei cambiamenti.    Figura 6 funzione model insertData parte 1  La funzione prima controlla se il dato esiste già, se è il caso allora non inserisce un nuovo dato ma modifica quello esistente inserendo la nuova quantità richiesta del relativo prodotto, altrimenti inserisce i nuovi dati.    Figura 7 funzione model insertData parte 2  Oltre alla funzione che inserisce o modifica i dati ce n’è una che prende il prodotto richiesto da inserire nel carrello e lo stampa, in modo che grazie a JavaScript si possa inserire nella pagina, come mostrato nell’immagine che segue.    Figura 8 funzione getCartProduct  Per inserire i dati, utilizzo la funzione “modifyCart” mostrata nel diario precedente a questo, ma con delle modifiche, perché oltre a mostrare i dati deve anche mostrare il numero di oggetti presenti all’interno del carrello, e se un oggetto è già presente non deve reinserirlo ma semplicemente modificare il numero di prodotti di quel tipo richiesti, come mostrato nell’immagine.    Figura 9 funzione modifica del carrello 1  Inizialmente la funzione non passa più tutti i dati ma viene richiamata ogni volta quindi l’unica cosa che fa è inserire l’oggetto che gli viene passato tramite argomento. Un primo controllo che si può vedere, viene fatto per assicurarsi che l’oggetto con quell’id non esiste già, e se è così crea l’oggetto come mostrato in quella e nella foto che segue, con le varie informazioni e Id.    Figura 10 funzione modifica del carrello 2    Figura 11 funzione modifica del carrello 3  Se invece l’oggetto esiste già viene aumentata la quantità di quest’ultimo e poi modificata all’interno del carrello.  Alla fine della funzione viene preso l’oggetto che mostra il carrello in alto a destra e viene scritto il numero dei prodotti inseriti.  Dopodiché ho messo a posto l’inserimento dei prodotti da parte dei venditori, facendo in modo che venga anche inserito nella tabella “vende” che, per l’appunto, un determinato negozio vende un certo prodotto.  Per faro ho inserito un nuovo select nella pagina di inserimento che permetta di scegliere quale negozio vende il determinato prodotto, come mostrato nell’immagine sottostante.    Figura pagina inserimento prodotti  L’implementazione di questo select mi ha preso un po’ di tempo, perché a differenza della categoria al submit non deve passare solo il dato che l’utente vede ma anche l’indirizzo e la città per poter avere la chiave completa, di conseguenza ho utilizzato anche in questo caso il metodo di concatenare in una stringa i dati e metterli come id dei vari campi, e una volta selezionato un dato viene preso l’id di questo e inserito nel select come per la categoria.  La funzione che prende i dati e li mette nl select è uguale a quella per la categoria ma fa riferimento a una tabella diversa.    Figura funzione di modifica del select dei negozi "setShop"  La funzione che si attiva una volta selezionato un negozio però è leggermente diversa da quella della categoria, perché va a prendere il valore, non dal paragrafo con id “current” che viene creato in automatico da template ma al punto che ha la classe “select”, ovvero il negozio selezionato, in modo che viene inserito nel select, non solo il nome scritto a schermo ma tutta la chiave codificata.  Una volta fatto, la funzione che inserisce i dati nel database va a inserire anche i dati nella tabella ponte che mostra quali negozi vedono quali prodotti, come mostrato nelle 2 immagini che seguono, la prima mostra la funzione controller e la seconda il model, per il model ho creato una nuova classe che faccia riferimento alla tabella “vende”.    Figura inserimento prodotto con controllo negozio    Figura funzione insrtData classe SellModel  Una volta finito questo mi manca, oltre al completamento della pagina de venditore, come spiegato nella sezione “Punto della situazione rispetto alla pianificazione”, devo fare in modo che se lo stesso prodotto con stesso prezzo e quantità esiste in 2 negozi e uno viene comprato non venga modificato per entrambi i negozi ma venga creata una nuova istanza per il negozio a cui è stato comprato con la quantità minore. |  |

|  |
| --- |
| Problemi riscontrati e soluzioni adottate |
| Ho messo a posto il problema riscontrato la scorsa lezione, come spiegato all’inizio della sezione precedente, e dopodiché non ho riscontrato particolari problemi. |

|  |
| --- |
| Punto della situazione rispetto alla pianificazione |
| Ho completato l’inserimento nel carrello.  Ho completato l’inserimento dei prodotti, in modo che venga anche segnalato chi li vende.  Mi manca la modifica dei prodotti già inseriti. |

|  |
| --- |
| Programma di massima per la prossima giornata di lavoro |
| Completare la modifica dei prodotti da parte dei venditori. |