

Web Systems Design and Architecture

Gabriele Musso - 0757473

| | |
|---------------------------------|----------|
| Descrizione progetto: | 2 |
| Funzioni richieste: | 2 |
| Chiunque: | 2 |
| Negoziante: | 2 |
| Admin: | 3 |
| Architettura: | 3 |
| Repository e file: | 3 |
| /model: | 3 |
| /servlet: | 3 |
| servlet/carta: | 3 |
| servlet/transazioni: | 3 |
| servlet/utente: | 3 |
| /utils: | 3 |
| webapp/assets: | 3 |
| webapp/WEB-INF/componenti: | 4 |
| webapp/WEB-INF/view: | 4 |
| webapp/WEB-INF: | 4 |
| Componenti software: | 4 |
| Carta: | 4 |
| Transazioni: | 4 |
| Utente: | 4 |
| Flusso delle operazioni: | 4 |

Descrizione progetto

Il sistema dovrà gestire delle carte prepagate ricaricabili dove chiunque potrà accedere alla verifica del credito residuo noto il numero della carta, i negozianti potranno effettuare ricariche e prelievi ed avere un report delle operazioni fatte, infine gli amministratori potranno creare nuove carte, bloccare o sbloccare una carta noto il suo numero e gestire gli utenti

Funzioni richieste

Chiunque

Accedere ad una pagina che permetta di verificare il credito residuo

Negoziante

Accedere a una pagina in cui può fare un addebito o una ricarica su una carta noto il numero
Generare report con le operazioni fatte

Admin

Creare una nuova carta
Bloccare/sbloccare una carta noto il numero
Gestire utenti (modifica tipo di account)

Architettura

È stata scelta un architettura MVC con l'uso di servlet e jsp, di seguito è descritto l'albero delle repository e dei file:

Repository e file

La descrizione utilizza un approccio di visita in profondità, quindi verranno discusse, in ordine alfabetico, tutte le repository arrivando fino ai file per poi passare alle successive.

/model

Qui sono definiti gli oggetti utente, carta e transazioni con relative query, il suffisso *Service* indica gli oggetti che permettono di recuperare i dati dal database

/servlet

Qui sono presenti tre repository che contengono le servlet per ogni tipo di oggetto

servlet/carta

Qui sono definiti gli endpoint con la logica relativa alla carta e alle operazioni associate

servlet/transazioni

Qui sono definiti gli endpoint con la logica relativa alle transazioni e alle operazioni associate

servlet/utente

Qui sono definiti gli endpoint con la logica relativa all'utente e alle operazioni associate

/utils

Qui sono definiti gli oggetti di utilità generale, tra le altre cose il connettore al database

webapp/assets

Qui sono presenti gli asset relativi al frontend, dato l'utilizzo di cdn sono presenti solo immagini

webapp/WEB-INF/componenti

Qui sono presenti componenti del frontend utili in più contesti

webapp/WEB-INF/view

Qui sono presenti tutte le pagine jsp

webapp/WEB-INF

Qui è presente il file di configurazione web.xml che presenta la configurazione per la pagina di index in quanto rimossa dal percorso predefinito

Componenti software

Java 17

Mysql 8

Apache Tomcat 10.1.12

JakartaEE 9.1

Modelli

Carta

La carta presenta i campi id (utilizzato come autoincrementale nel db), numeroCarta (che di fatto è l'identificativo utilizzato nel sistema), attiva (che definisce se la carta è attiva o meno) e creditoResiduo.

Transazioni

Le transazioni presentano i campi id (utilizzato come autoincrementale nel db), utente (email dell'operatore che ha effettuato l'operazione), timestamp (millisecondi dal tempo 0 unix), operazione (somma aggiunta o sottratta alla carta) e numeroCarta.

Utente

L'utente presenta i campi id (utilizzato come autoincrementale nel db), nome, cognome, email, password e ruolo.

Flusso delle operazioni

Un utente che non ha fatto accesso può comunque verificare il credito residuo di una carta noto il numero e questo è possibile attraverso l'accesso alla dashboard che mostrerà solo quella possibilità.

Un utente che ha fatto l'accesso ed è di tipo utente (quindi con privilegi minimi) vedrà una schermata del tutto simile a quella dell'utente che non ha effettuato accesso eccetto per la presenza del tasto logout

Un utente che ha fatto l'accesso ed è di tipo negoziante vedrà la barra laterale popolata dalle voci verifica *credito*, *report* e *effettua operazione* oltre al *logout*.

La voce verifica credito è la stessa che utilizza l'utente non registrato, mentre report è legata all'utenza e presenta i movimenti effettuati dal negoziante, infine effettua operazione permette di incrementare o decrementare il credito di una carta noto il numero della stessa.

Un utente che ha effettuato l'accesso ed è di tipo admin vedrà tutte le opzioni del negoziante più la modifica del ruolo dell'utente, modifica dello stato di una carta e verifica dello stato della stessa. Il primo componente permette di modificare il ruolo che un utente ha nel sistema, il secondo permette di modificare lo stato della carta e l'ultimo lo verifica.