

Car parts

ANALISI DEI REQUISITI

Francesco Campagna
A.A 2023/2024 | 230750

Sommario

ANALISI DEI REQUISITI UTENTE	2
DIAGRAMMA DEI CASI D'USO	3
SCHEDE DEI CASI D'USO	4
DIAGRAMMA DEL MODELLO DI DOMINIO	9
SSD DI SISTEMA	11
CONTRATTI DELLE OPERAZIONI	13
ARCHITETTURA SOFTWARE	14

PROGETTO – Car parts.

1. ANALISI DEI REQUISITI UTENTE

Car parts è un applicazione desktop per l'acquisto online di componenti originali e compatibili relativi al mondo dell'automobilismo.

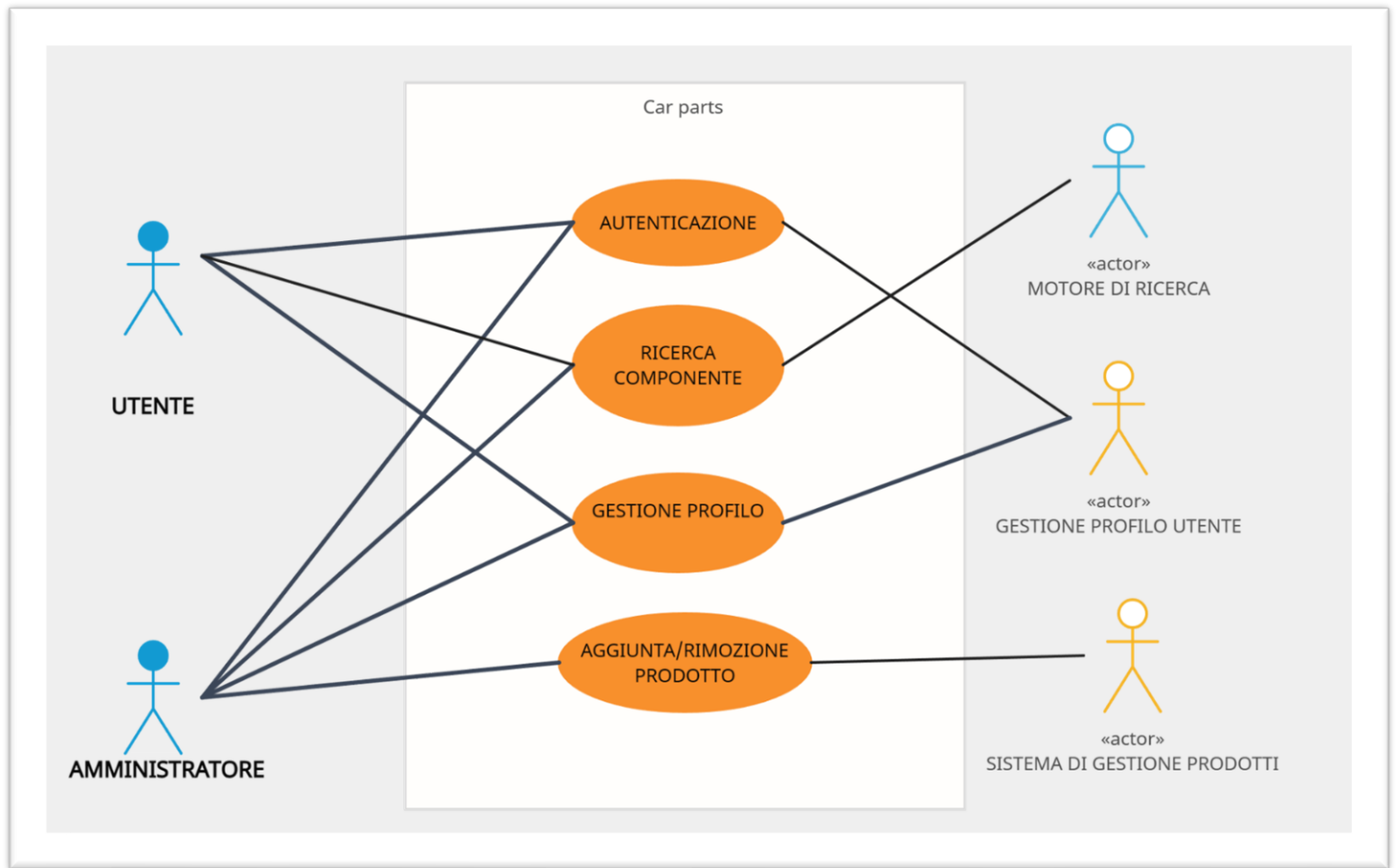
La finalità dell'utente, di conseguenza l'obiettivo del sistema, è quello di consentire la ricerca del componente attraverso apposita barra di ricerca o tramite la suddivisione in categorie dei prodotti disponibili.

Per semplificare l'esperienza dell'utente, quest'ultimo avrà la possibilità di posticipare l'acquisto inserendo il prodotto nel carrello o nella propria wishlist; per fare ciò l'utente ha bisogno di un account personale che può essere creato nell'apposita sezione dell'app. Nella sezione dedicata l'utente avrà la possibilità di modificare i propri dati personali e indirizzi di spedizione anche in una fase successiva alla registrazione, riscattare buoni e visualizzare il proprio dettagliato storico ordini, comprensivo di totale ordine, nome e numero di componenti ordinati. L'interfaccia presenterà un insieme di prodotti, con apposita schermata dedicata ottenuta selezionando un componente, la quale mostrerà la foto del prodotto, una descrizione esaustiva, i feedback degli utenti, il prezzo, il venditore e articoli correlati a quello selezionato.

Gli utenti amministratori avranno la possibilità di aggiungere, rimuovere e aggiornare la disponibilità dei prodotti, nonché di introdurre nuovi buoni da riscattare.

L'applicazione inoltre include pagine di supporto, le quali forniscono linee guida dell'applicazione e contatti per segnalazioni e richieste personalizzate.

2. DIAGRAMMA DEI CASI D'USO



3. SCHEDE DEI CASI D'USO

3.1 Autenticazione

Caso d'uso: Autenticazione

Portata: Applicazione Car parts

Livello: Obiettivo utente

Attore Primario: Utente

Parti Interessate e Interessi:

- Utente: Desidera realizzare l'autenticazione per beneficiare dei servizi offerti.
- Sistema: È responsabile di gestire e convalidare le informazioni fornite dall'utente.

Scenario principale di successo:

1. L'utente vuole autenticarsi al sistema.
2. Il sistema chiede di inserire i dati.
3. L'utente fornisce le informazioni richieste.
4. Il sistema verifica la correttezza dei dati inseriti.

Estensioni (Flussi alternativi):

1a. L'utente vuole registrarsi

1. Il sistema chiede all'utente di inserire i campi necessari per la registrazione.
2. L'utente inserisce i dati personali
3. Il sistema verifica la validità dei dati inseriti
4. Il sistema crea un nuovo profilo per l'utente

2a. Password dimenticata

1. L'utente clicca su password dimenticata per recuperarla.
2. Il sistema chiede all'utente di fornire l'e-mail associata all'account.
3. Il sistema verifica la validità dell'e-mail.
4. Il sistema invia per e-mail la procedura di recupero password.

4a. I dati inseriti risultano non validi:

1. Il sistema identifica con precisione quali sono i dati che non soddisfano i requisiti minimi (e-mail non disponibile, password debole, campi obbligatori mancanti).
2. Mostra un messaggio di errore dettagliato e fornisce suggerimenti per correggere ciascun dato non valido.
3. Richiede all'utente di apportare correzioni in base alle indicazioni fornite.
4. Si ritorna al punto 3 per una nuova verifica dei dati.

4b. L'utente non riempie tutti i campi:

1. Il sistema segnala l'errore e richiede l'inserimento dei dati mancanti.
2. L'utente controlla e riempie campi precedentemente non compilati.
3. Si ritorna al punto 2 per completare la registrazione.

4c. L'applicazione subisce un crash:

1. Il sistema rileva il crash e visualizza un messaggio di errore all'utente, segnalando il problema tecnico.
2. Il sistema consiglia all'utente di riavviare l'applicazione per ripristinare il corretto funzionamento.
3. Si ritorna al punto 2 per completare la registrazione.

4d. Account bloccato

1. Il sistema rileva troppi tentativi errati e blocca temporaneamente l'account mostrando un messaggio d'errore.
2. Il sistema guida l'utente attraverso un processo per sbloccare l'account.

Requisiti speciali:

- L'utente deve poter visualizzare la schermata di autenticazione con una singola interazione con l'interfaccia.

Requisiti non funzionali:

- Usabilità: l'interfaccia deve essere progettata in modo intuitivo e facilitando la navigazione degli utenti.
- Affidabilità: l'applicazione deve essere affidabile, minimizzando i tempi di inattività e garantendo un'operatività costante.

Frequenze di ripetizione:

- Ogni qual volta un utente desidera autenticarsi per poter usufruire dei servizi.

Problemi aperti:

- Possibilità di migliorare la verifica dei dati inseriti per evitare errori durante la registrazione.

3.2 Acquisto

Caso d'uso: Acquisto

Portata: Applicazione Car parts

Livello: Obiettivo utente

Attore Primario: Utente

Parti Interessate e Interessi:

- Utente: Desidera acquistare i componenti.
- Sistema: È responsabile di gestire la vendita.

Scenario principale di successo:

1. L'utente ricerca e aggiunge al carrello i/il prodotto/i da acquistare.
2. L'utente procede con la conferma dell'ordine.
3. Il sistema chiede di inserire i dati per il pagamento e la spedizione.
4. L'utente inserisce i dati richiesti.
5. Il sistema verifica i dati inseriti.
6. Il sistema comunica il completamento dell'ordine.

Estensioni (Flussi alternativi):

1a. Ricerca senza risultati

1. Nessun risultato corrisponde alla ricerca dell'utente.
2. Il sistema mostra un messaggio che informa l'utente della mancanza di risultati.
3. Il sistema propone all'utente prodotti correlati.

2a. Il prodotto non è disponibile

1. Il sistema informa il cliente sulla mancata disponibilità del componente.
2. Il sistema propone all'utente prodotti correlati.

3a. L'utente inserisce il buono da riscattare

1. L'utente inserisce il codice del buono.
2. Il sistema verifica il valore e la validità del codice e informa l'utente.
3. Si continua al punto 4.

3b. Ritiro in negozio fisico

1. L'utente sceglie l'opzione di ritiro in un negozio fisico.
2. Il sistema fornisce un elenco di negozi disponibili.
3. L'utente seleziona il negozio e riceve le istruzioni per il ritiro.

4a. L'utente vuole annullare l'ordine

1. L'utente annulla l'ordine.
2. Il sistema chiede una conferma aggiuntiva per l'annullamento.
3. Il sistema annulla l'ordine.

5a. L'indirizzo di spedizione non è valido:

1. Il sistema identifica l'indirizzo di spedizione errato.
2. Richiede all'utente di apportare correzioni in base alle indicazioni fornite.

6a. Saldo carta insufficiente

1. Il sistema comunica il saldo insufficiente.
2. Si ritorna al punto 4.

6b. Ritardo nella spedizione

1. Dopo la conferma dell'ordine il sistema comunica un ritardo nella spedizione.
2. Il sistema fornisce una nuova stima della data di consegna.
3. L'utente sceglie se attendere o annullare l'ordine.

Requisiti speciali:

- L'organizzazione dell'interfaccia deve spingere l'utente a confermare l'ordine.
- Il pagamento deve essere completato entro 3 passaggi.
- Performance: L'applicazione deve garantire tempi di risposta rapidi durante tutte le fasi dell'acquisto.

Frequenza di ripetizione:

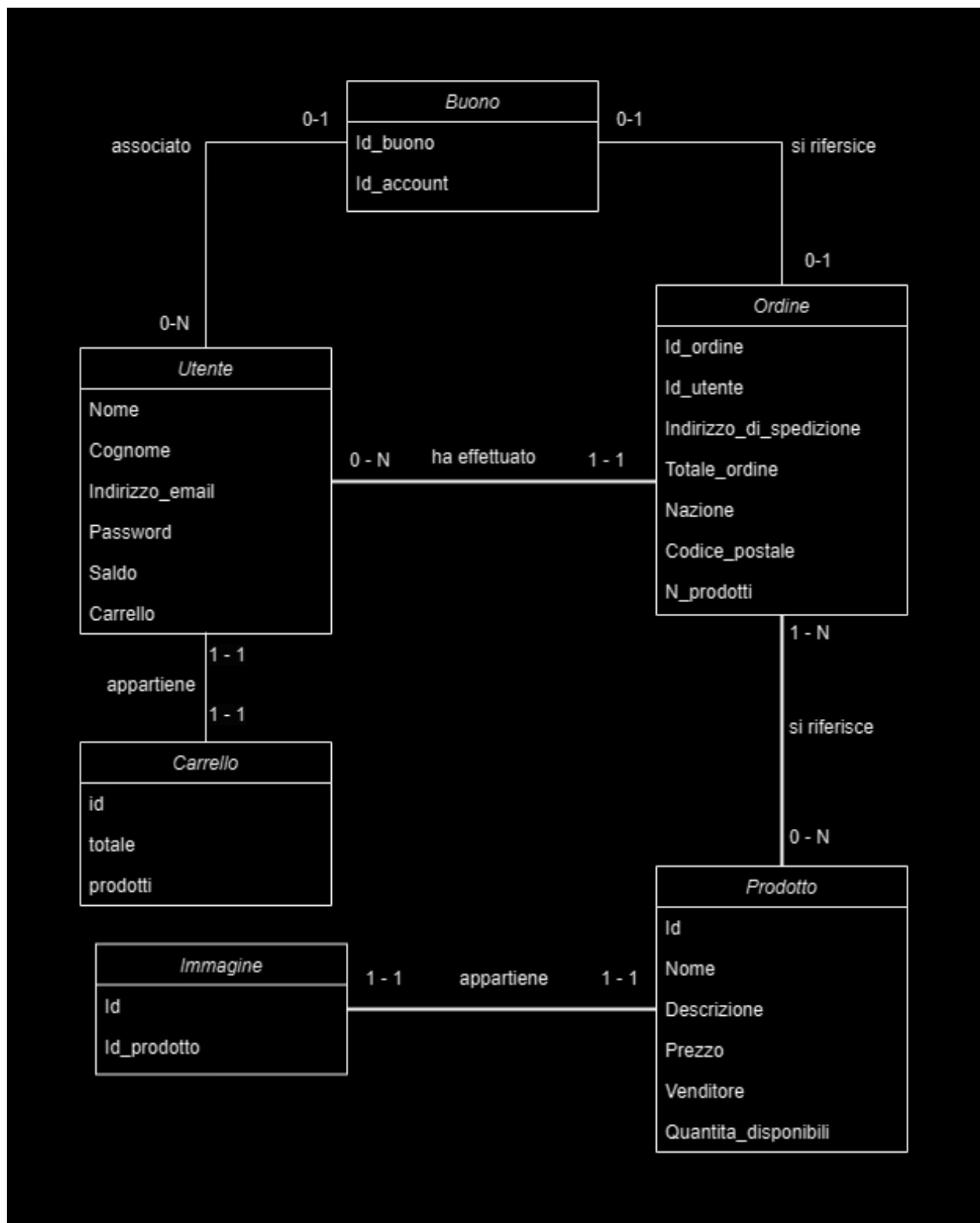
- Ogni qual volta un utente effettua un ordine.

Problemi aperti:

- L'utente deve poter salvare la carta di credito per acquisti futuri.

4. DIAGRAMMA DEL MODELLO DI DOMINIO

4.1 Diagramma

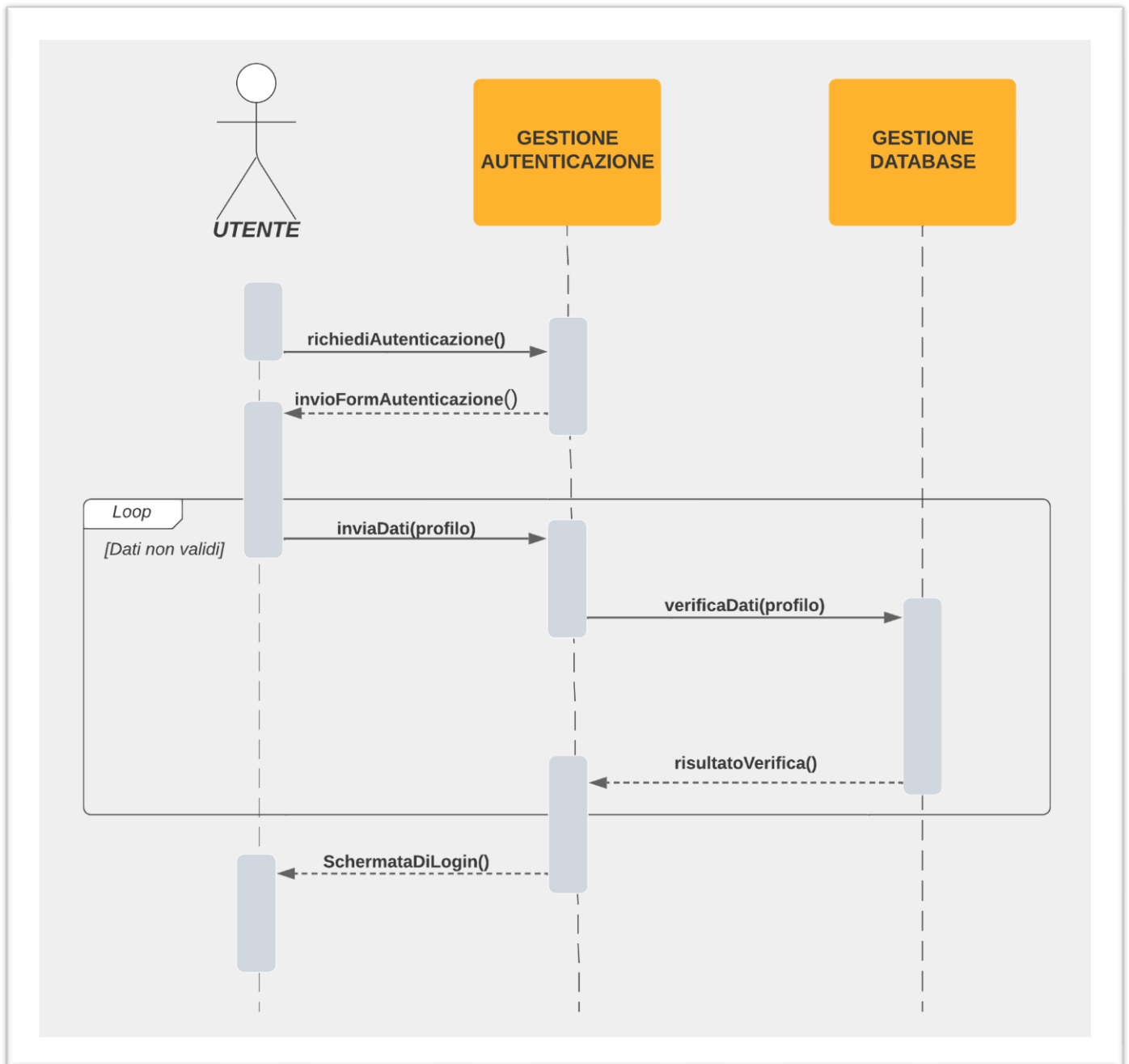


4.2 Descrizione del modello

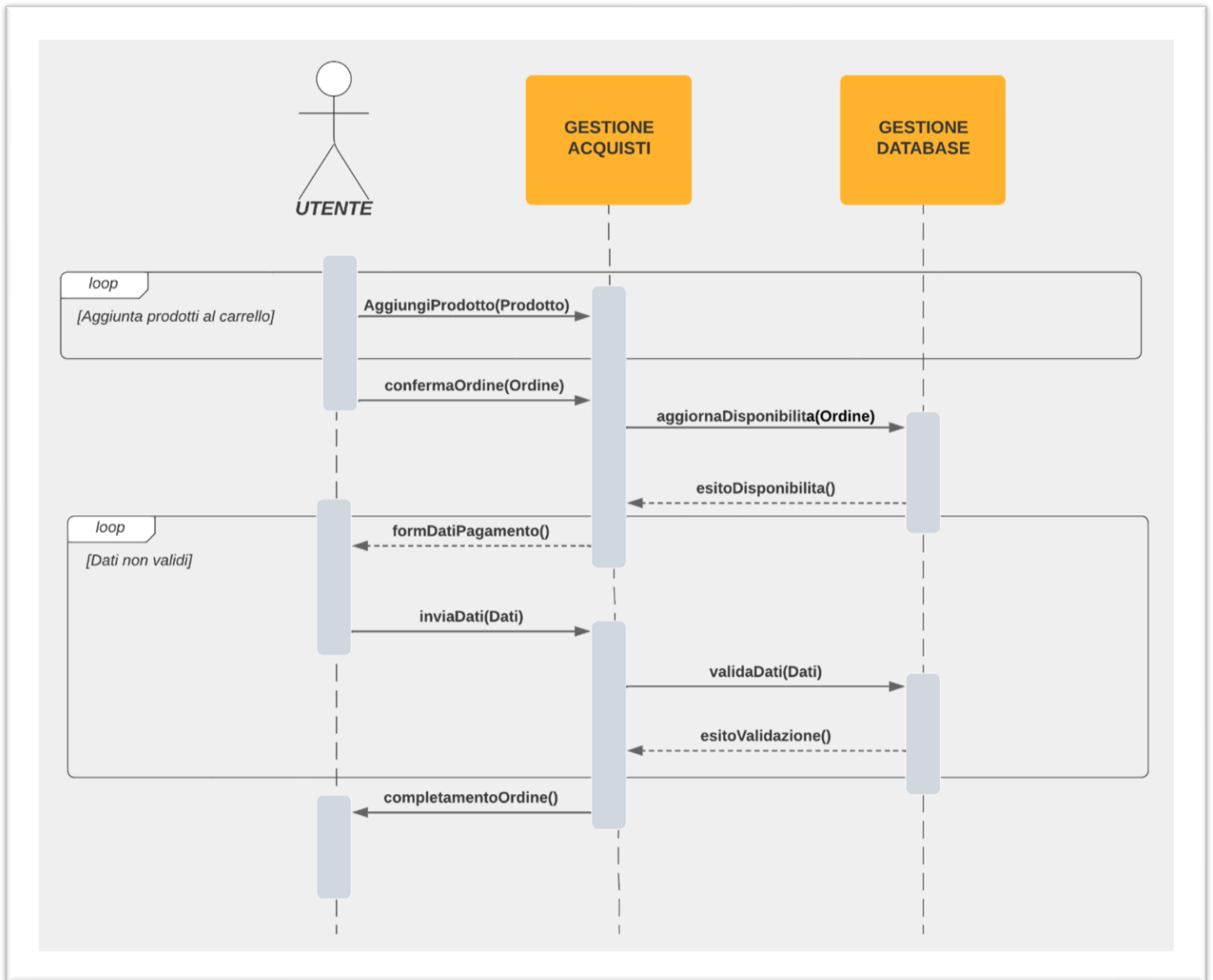
Ogni utente possiede al massimo un account. Ad ogni utente appartiene un carrello. Ogni utente può effettuare nel tempo zero o più ordini rappresentati da un id identificativo, può riscattare zero o più buoni, anch'essi identificati da un id, e associare un buono in fase di acquisto ad un solo ordine. Ad ogni ordine è associato uno o più prodotti identificato/i ciascuno da un id. Ad ogni prodotto appartiene una ed una sola immagine rappresentante il prodotto.

5. SSD DI SISTEMA

5.1 Autenticazione



5.2 Acquisto



6. Contratti delle operazioni

6.1 CO1: Inserisci prodotto nel carrello

Operazione: AggiungiProdotto(Prodotto: Product)

Riferimenti: caso d'uso: Acquisto

Pre-condizioni: L'utente ha effettuato il login.

Post-condizioni:

- È stata creata un'istanza c rappresentante il carrello.
- Il prodotto p selezionato dall'utente viene aggiunto a c.
- Il c.totale è stato aggiornato.

6.2 CO2: Verifica autenticazione

Operazione: InviaDati(Profilo: Profile)

Riferimenti: caso d'uso: Autenticazione

Pre-condizioni: L'utente ha richiesto l'autenticazione.

Post-condizioni:

- È stata creata un'istanza p rappresentante il profilo.
- Il profilo p viene ricevuto dal gestore autenticazione ga.
- ga ha verificato i dati tramite il database db.

7. ARCHITETTURA SOFTWARE

UI LAYER:

Si occupa della visualizzazione dell'interfaccia grafica basandosi sulla libreria JavaFX e personalizzandola con CSS.

APPLICATION:

Riguarda i controller che comunicano con l'interfaccia grafica, gestiscono gli input degli utenti e a loro volta coordinano questi input attraverso i servizi di sistema corrispondenti.

DOMAIN:

Si occupa di immagazzinare e gestire i dati relativi all'applicazione e quindi al modello di dominio.

