**Università degli Studi di Salerno**

Corso di Ingegneria del Software

**UNISA-TCG**

**Requirements Analysis Document**



Data: 28/10/2025

|  |  |
| --- | --- |
| Progetto: UNISA-TCG | Versione: 1.0 |
| Documento: Problem Statement | Data: 14/10/2025 |

Coordinatore del progetto:

|  |  |
| --- | --- |
| **Nome** | **Matricola** |
| Sepe Giuseppe | 0512119386 |

**Partecipanti:**

|  |  |
| --- | --- |
| **Nome** | **Matricola** |
| Senatore Francesco | 0512120568 |
| Sepe Giuseppe | 0512119386 |
| Sullo Diego | 0512119455 |

|  |  |
| --- | --- |
| **Scritto da:** | Senatore Francesco, Sepe Giuseppe, Sullo Diego |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versione** | **Descrizione** | **Autore** |
| **14/10/2025** | **1.0** | **Problem Statement** | **Francesco Senatore, Diego Sullo, Giuseppe Sepe** |

**Indice**

[3](#_TOC_250001)

1. Introductin 3
2. Scenarios
   1. Ricerca carta rara Pokémon 4
   2. Aggiunta al carrello e modifica quantità 4
   3. Checkout con carta 4
   4. Gestione ordine effettuati 4
   5. Gestione catalogo e ordini 4
3. Non-functional requirements 5

        3.1 Usability 5  
    3.2 Reliability 5  
        3.3 Performance 6  
        3.4 Supportability 6  
        3.5 Implementation 6  
        3.6 Interface 6  
        3.7 Packaging 6  
        3.8 Legal 6

1. Use cases 7
2. Mock-ups 11

***1. Introduction***

Il presente documento definisce e analizza i requisiti del sistema **UNISA-TCG**, una piattaforma web dedicata alla compravendita e gestione di carte collezionabili appartenenti ai principali giochi di carte (Pokémon, Yu-Gi-Oh!, Magic: The Gathering). Il progetto nasce dall’esigenza di superare la frammentazione attuale del mercato, caratterizzato da cataloghi disomogenei, informazioni incomplete e processi di acquisto poco uniformi. L’obiettivo è realizzare un ambiente unificato, affidabile e facilmente utilizzabile, che consenta agli utenti di cercare, filtrare, acquistare e recensire carte o accessori in modo sicuro e intuitivo, e ai gestori di amministrare cataloghi, ordini e prezzi da un’interfaccia centralizzata. Lo sviluppo, basato sul Problem Statement fornito dal committente e su un’analisi di fattibilità rispetto a piattaforme esistenti (es. CardMarket, TCGPlayer, eBay), prevede un’architettura web responsive realizzata con **Java (Servlet/JSP)** e **MySQL**, eseguita su **Apache Tomcat**. Il progetto si propone di garantire usabilità, sicurezza, scalabilità e prestazioni elevate, con tempi di risposta inferiori a un secondo nelle operazioni di ricerca e un uptime minimo del 99%. Il successo del sistema sarà misurato dalla sua capacità di supportare l’intero flusso d’acquisto senza errori, mantenere la protezione dei dati personali secondo il GDPR e offrire un’esperienza d’uso coerente e soddisfacente su tutti i dispositivi e browser supportati.

Il presente documento, denominato Requirements Analysis Document (RAD), descrive e analizza i requisiti funzionali e non funzionali del sistema **UNISA-TCG**, una piattaforma web dedicata alla compravendita e gestione di carte collezionabili dei principali giochi di carte da collezione (Pokémon, Yu-Gi-Oh!, Magic: The Gathering). Il progetto è stato avviato per rispondere alla frammentazione del mercato dei TCG, caratterizzato da cataloghi eterogenei, informazioni incomplete e processi di acquisto poco standardizzati. L’obiettivo è la realizzazione di un ambiente unico, sicuro e intuitivo che consenta agli utenti di ricercare, filtrare, acquistare e recensire prodotti con facilità, e ai gestori di amministrare cataloghi, ordini e prezzi in modo efficiente.

Lo sviluppo del sistema si fonda sul Problem Statement redatto dal committente e tiene conto di un’analisi comparativa con piattaforme esistenti come CardMarket, TCGPlayer ed eBay, evidenziandone limiti e opportunità di miglioramento. La piattaforma sarà implementata come applicazione web responsive, basata su **Java (Servlet/JSP)** e **MySQL**, distribuita su **Apache Tomcat**, e accessibile da dispositivi desktop e mobili.

Gli obiettivi principali del progetto comprendono l’usabilità dell’interfaccia, la sicurezza delle transazioni, la scalabilità dell’infrastruttura e l’efficienza delle operazioni di ricerca e gestione. Il successo del sistema sarà determinato dalla capacità di supportare il flusso completo di acquisto senza errori, garantire la protezione dei dati personali nel rispetto del GDPR, mantenere un tempo di risposta inferiore a un secondo per le principali operazioni e assicurare un uptime minimo del 99%, offrendo un’esperienza coerente e soddisfacente per l’utente finale.

***2. Scenarios***

* 1. **Ricerca carta rara Pokémon**

**Scenario Concreto:** Un acquirente cerca “Charizard”, applica il filtro “Carte singole” e apre la scheda prodotto più rilevante.

**Attori partecipanti: Acquirente**

* 1. **Aggiunta al carrello e modifica quantità**

**Scenario Concreto:** L’acquirente aggiunge un booster “Jungle” e un accessorio “Bustine protettive”, in seguito nel carrello porta la quantità del booster da 1 a 3 e rimuove l’accessorio.

**Attori partecipanti: Acquirente**

* 1. **Checkout con carta**

**Scenario Concreto:** L’acquirente inserisce indirizzo, esegue il pagamento, e riceve conferma via email con link di tracking.

**Attori partecipanti: Acquirente**, **Payment Service Provider (simulato)**, **Servizio email/notifiche**

* 1. **Gestione ordine effettuati**

**Scenario Concreto:** L’acquirente apre “I miei ordini”, contenente tutti gli ordini effettuati da lui e i vari dettagli di ogni ordine (prodotti acquistati, spedizione, indirizzo di fatturazione)

**Attori partecipanti: Acquirente**, **Servizio email/notifiche**, **Corriere (simulato)**

* 1. **Gestione catalogo e ordini**

**Scenario Concreto:** Il gestore aggiorna il prezzo di una serie di carte, imposta la quantità di un booster a 0 (esaurito) e visualizza gli ordini del giorno.

**Attori partecipanti: Gestore**

**2.6 Registrazione di un Nuovo Acquirente**

**Scenario Concreto**: Un utente ospite accede alla piattaforma per la prima volta, decide di registrarsi e inserisce i propri dati personali. Il sistema verifica la disponibilità dell'indirizzo email e completa la registrazione.

**Attori partecipanti**: Acquirente, Servizio email/notifiche

**2.7 Pubblicazione di una Recensione (Dopo l'Acquisto)**

**Scenario Concreto**: Un Acquirente, dopo aver ricevuto l'ordine, torna sulla scheda prodotto e pubblica una recensione e una valutazione a stelle sulle condizioni di una carta singola.

**Attori partecipanti**: Acquirente

**2.8 Richiesta di Reset Password**

**Scenario Concreto**: Un Acquirente dimentica la password per accedere al proprio account. Inserisce il proprio indirizzo email nella sezione di recupero credenziali. Il Sistema verifica l'email, invia un link temporaneo univoco all'indirizzo email e ne riceve conferma via notifica.

**Attori partecipanti**: Acquirente, Servizio email/notifiche

***3. Nonfunctional requirements***

#### **3.1 Usability**

Il sistema deve offrire un’interfaccia intuitiva, coerente e facilmente navigabile per tutti i tipi di utenti (acquirenti, gestori, amministratori).

L’interfaccia utente sarà **responsive**, adattandosi automaticamente a diverse risoluzioni (desktop, tablet, smartphone).

#### **3.2 Reliability**

Il sistema deve assicurare un funzionamento stabile e continuo nel tempo.

In caso di errore o guasto, il sistema dovrà fornire **messaggi d’errore significativi** e gestire i rollback delle transazioni per prevenire perdite di dati.

Sono previsti **backup periodici** del database per garantire il ripristino dei dati in caso di incidente.

#### **3.4 Supportability**

Il sistema deve essere facilmente aggiornabile, manutenibile e monitorabile. Il codice sarà **modulare e documentato**, seguendo standard Java e convenzioni di naming.

Il database dovrà essere **normalizzato** e supportare l’estensione di nuove tabelle (es. nuovi giochi o categorie di prodotti).

#### **Implementation**

#### **Front-end:** sviluppato come applicazione web responsive in **HTML, JavaScript**.

#### **Back-end:** implementato in **Java (Servlet/JSP),** con **Tomcat** come application server.

#### **Database:** **MySQL**, con connessione tramite JDBC.

#### **Integrazioni esterne:** servizi simulati per pagamento e notifiche e-mail.

#### **Interface**

**Interfaccia utente:** interazione tramite browser web, con componenti grafici chiari e coerenti.

**Compatibilità browser:** Chrome, Firefox, Safari, Edge (ultime versioni).

#### **Packaging**

Il sistema sarà distribuito come **web application (WAR)** installabile su un server Tomcat. Il pacchetto includerà file di configurazione, codice sorgente compilato e script SQL di inizializzazione del database.

#### **Legal**

Tutti i dati personali degli utenti devono essere trattati nel rispetto del **Regolamento UE 2016/679 (GDPR).** Le immagini e i loghi delle carte collezionabili (Pokémon, Yu-Gi-Oh!, MTG) saranno utilizzati **solo a scopo dimostrativo e non commerciale**, nel rispetto dei diritti d’autore.

***4. Use Cases***

**UC1: Ricercare Prodotti nel Catalogo**

| **Descrizione** | **Dettagli** |
| --- | --- |
| **Use case name** | Ricercare Prodotti nel Catalogo |
| **Participating Actors** | Acquirente |
| **Entry Condition** | Il Catalogo è disponibile e popolato. |
| **Flow of Events** | L'Acquirente accede alla pagina di ricerca. L'Acquirente inserisce termini di ricerca e/o applica filtri (set, lingua, rarità, condizioni, prezzo)3333. Il Sistema visualizza un elenco di prodotti corrispondenti. L'Acquirente seleziona un prodotto e visualizza la scheda dettagliata4. |
| **Exit Condition** | L'Acquirente ha visualizzato la scheda prodotto o è tornato alla pagina di ricerca. |
| **Special Requirements** | I filtri devono essere efficaci e il catalogo deve essere unificato e standardizzato5555. Le prestazioni devono garantire tempi di caricamento rapidi anche con cataloghi ampi6 |

**UC2: Gestire il Carrello**

| **Descrizione** | **Dettagli** |
| --- | --- |
| **Use case name** | Gestire il Carrello |
| **Participating Actors** | Acquirente |
| **Entry Condition** | L'Acquirente ha aggiunto almeno un prodotto al carrello8. |
| **Flow of Events** | L'Acquirente accede alla schermata del Carrello. L'Acquirente modifica la quantità di un prodotto (es. da 1 a 3 booster)9. L'Acquirente rimuove un prodotto dal carrello (es. un accessorio)10. Il Sistema aggiorna il totale provvisorio dell'ordine. |
| **Exit Condition** | Il carrello riflette le modifiche desiderate dall'Acquirente, ed è pronto per il checkout. |
| **Special Requirements** | Il sistema deve gestire la disponibilità degli articoli durante l'aggiunta al carrello. |

**UC3: Effettuare il Checkout e il Pagamento**

| **Descrizione** | **Dettagli** |
| --- | --- |
| **Use case name** | Effettuare il Checkout e il Pagamento |
| **Participating Actors** | Acquirente, Payment Service Provider (simulato), Servizio email/notifiche 11 |
| **Entry Condition** | L'Acquirente è autenticato (o procede come ospite) , il carrello non è vuoto e il metodo di pagamento è valido. |
| **Flow of Events** | L'Acquirente inserisce i dati di indirizzo di spedizione e fatturazione12. L'Acquirente seleziona il metodo di pagamento (carta di credito o PayPal)13. Il Sistema delega la validazione dei dati sensibili al Payment Service Provider (PSP)14. Il PSP (simulato) conferma il successo della transazione. Il Sistema finalizza l'ordine, invia la conferma con il link di tracking via email/notifiche15. |
| **Exit Condition** | L'ordine è completato, la transazione è registrata e l'Acquirente riceve conferma. |
| **Special Requirements** | Tutte le transazioni e i dati sensibili devono essere protetti tramite HTTPS e crittografia16. |

**UC4: Visualizzare e Gestire gli Ordini del Sistema**

| **Descrizione** | **Dettagli** |
| --- | --- |
| **Use case name** | Visualizzare e Gestire gli Ordini del Sistema |
| **Participating Actors** | Gestore 21 |
| **Entry Condition** | Il Gestore è autenticato nel Pannello Amministratore. |
| **Flow of Events** | Il Gestore accede alla sezione "Ordini" nel Pannello Amministratore. Il Gestore visualizza l'elenco degli ordini recenti (es. gli ordini del giorno)22. Il Gestore seleziona un ordine per visualizzarne tutti i dettagli (prodotti acquistati, indirizzo, stato). Il Gestore aggiorna lo stato di spedizione dell'ordine (simulato). |
| **Exit Condition** | Il Gestore ha completato la revisione e la gestione degli ordini necessari. |
| **Special Requirements** | Il sistema deve implementare un pannello amministratore per la gestione degli ordini23. |

**UC5: Gestire il Catalogo**

| **Descrizione** | **Dettagli** |
| --- | --- |
| **Use case name** | Gestire il Catalogo |
| **Participating Actors** | Gestore 17 |
| **Entry Condition** | Il Gestore è autenticato nel Pannello Amministratore. |
| **Flow of Events** | Il Gestore ricerca o seleziona un prodotto (es. una serie di carte o un booster). Il Gestore aggiorna il prezzo del prodotto18. Il Gestore modifica la disponibilità del prodotto (es. imposta la quantità a 0 per un booster esaurito)19. Il Sistema aggiorna il Catalogo e sincronizza i dati. |
| **Exit Condition** | I dati dei prodotti (prezzo, quantità, ecc.) nel catalogo sono stati aggiornati. |
| **Special Requirements** | Il Pannello Amministratore deve consentire la gestione dei contenuti, degli utenti e delle transazioni20. |

**UC6: Registrare un Nuovo Account**

| **Descrizione** | **Dettagli** |
| --- | --- |
| **Use case name** | Registrare un Nuovo Account |
| **Participating Actors** | Acquirente, Servizio email/notifiche |
| **Entry Condition** | L'utente si trova sulla pagina di registrazione. |
| **Flow of Events** | L'Acquirente inserisce i dati personali per la registrazione111. Il Sistema verifica l'unicità e la validità dell'indirizzo email. Il Sistema crea il nuovo account utente2. Il Servizio email/notifiche invia un'email di conferma all'Acquirente. |
| **Exit Condition** | L'Acquirente è registrato e autenticato sulla piattaforma3. |
| **Special Requirements** | Il sistema deve permettere la registrazione e l'autenticazione degli utenti4. |

**UC7: Pubblicare Recensioni e Valutazioni**

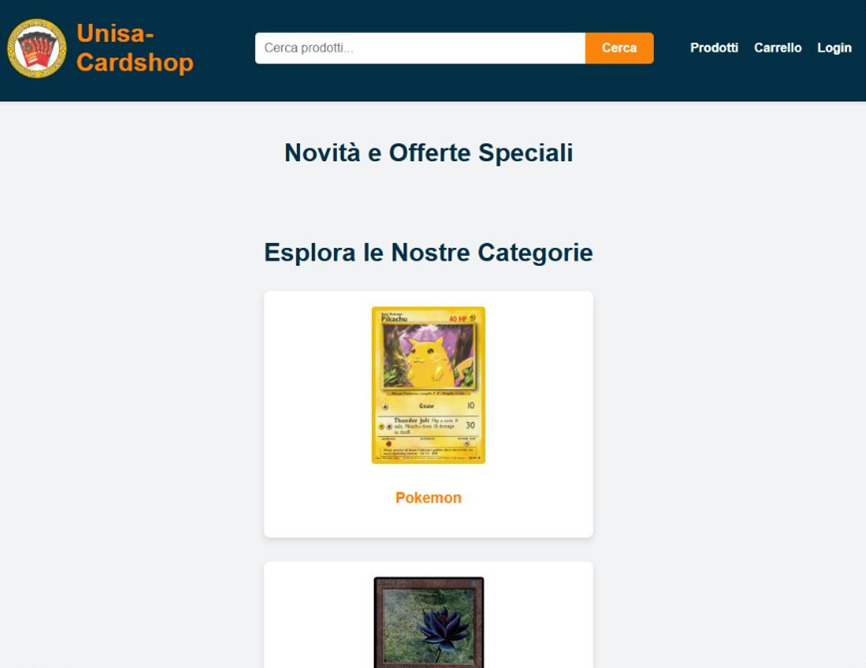
| **Descrizione** | **Dettagli** |
| --- | --- |
| **Use case name** | Pubblicare Recensioni e Valutazioni |
| **Participating Actors** | Acquirente |
| **Entry Condition** | L'Acquirente è autenticato e ha ricevuto il prodotto che intende recensire. |
| **Flow of Events** | L'Acquirente accede alla scheda prodotto. L'Acquirente inserisce una recensione testuale e una valutazione a stelle5. Il Sistema verifica la validità del contributo. Il Sistema registra la recensione e aggiorna la valutazione media del prodotto. |
| **Exit Condition** | La recensione e la valutazione sono state registrate dal Sistema e associate al prodotto6. |
| **Special Requirements** | Il sistema deve permettere recensioni e valutazioni per utenti e prodotti7. |

**UC8: Richiedere il Reset della Password**

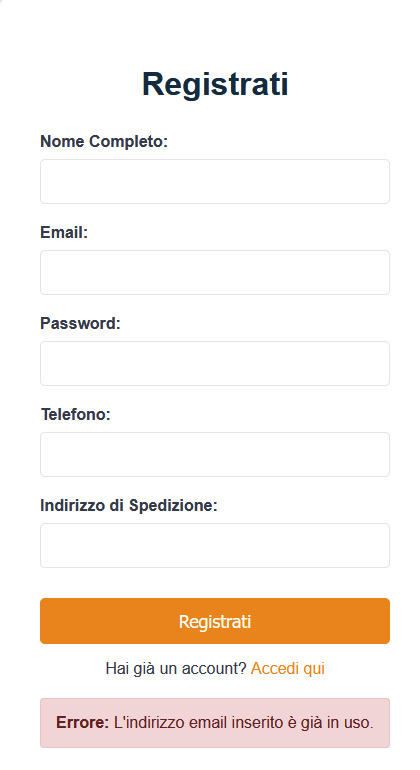
| **Descrizione** | **Dettagli** |
| --- | --- |
| **Use case name** | Richiedere il Reset della Password |
| **Participating Actors** | Acquirente, Servizio email/notifiche |
| **Entry Condition** | L'Acquirente si trova nella sezione di recupero credenziali. |
| **Flow of Events** | L'Acquirente inserisce il proprio indirizzo email. Il Sistema verifica che l'email corrisponda a un account registrato. Il Sistema genera un link temporaneo univoco. Il Servizio email/notifiche invia il link per il reset della password all'Acquirente. L'Acquirente riceve una conferma dell'invio via notifica. |
| **Exit Condition** | È stata inviata un'email contenente un link valido per la reimpostazione della password. |
| **Special Requirements** | Le transazioni e i dati sensibili devono essere protetti. |

***5. Mock-ups***

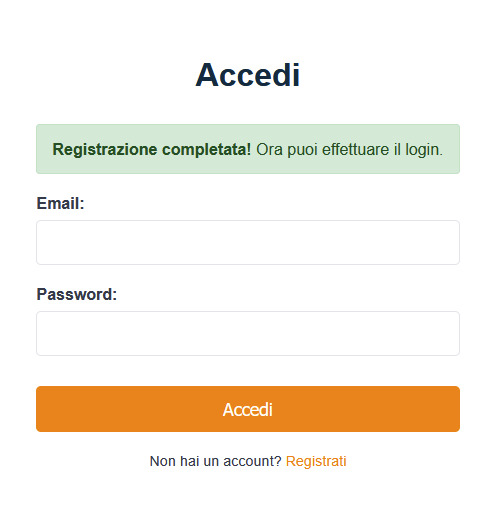
**Home Page**



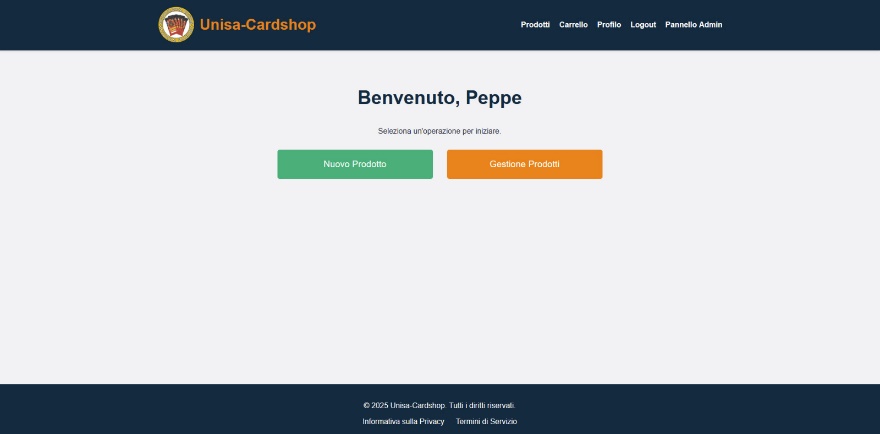
**Registrazione di un utente (con messaggio di errore)**

****

**Pagina di accesso**

****

**Vista da Amministratore**

****