

# Requisiti

Versione	Data	Descrizione
1.0	26 Novembre 2025	Prima versione del documento Requisiti

## 1. Qualità del software

Per garantire la qualità del software utilizzeremo la tassonomia di McCall che definisce tre categorie di fattori di qualità; noi ci concentreremo solo su alcuni:

1. Funzionamento del prodotto:
  - Correttezza: dovrà coprire tutti i requisiti elicitati
    - corretto utilizzo del Product Backlog con tutte le User Stories
    - i requisiti dovranno avere criteri di accettazione chiari
  - Affidabilità: dovrà funzionare senza crash
    - gestione delle eccezioni
    - utilizzo piattaforma di logging
  - Usabilità: dovrà essere semplice ed intuitiva da usare
    - GUI facile da interpretare e utilizzare
2. Revisione del prodotto:
  - Manutenibilità: dovrà essere facile da mantenere
    - documentazione del codice e naming curati
  - Testabilità: dovrà avere test sulle regole di conteggio punti
    - test JUnit per simulare le giocate
3. Transizione del prodotto:
  - Portabilità: dovrà funzionare su diversi tipi di client
    - JVM per astrarre l'architettura del calcolatore

## 2. Ingegneria dei requisiti

### Elicitazione dei requisiti

L'elicitazione dei requisiti è avvenuta tramite:

- Analisi delle attività a partire dall'analisi delle regole di gioco
- Brainstorming tra i membri del team di sviluppo
- Descrizioni in linguaggio naturale

Ovviamente il nostro team simulerà i bisogni e le richieste del cliente e impersonificherà il Product Owner.

### Specificazione dei requisiti

Per la specifica dei requisiti utilizzeremo lo standard **IEEE 830** che ne definisce la struttura:

1. Introduzione
  - 1.1. Obiettivo: la specifica dei requisiti ha il compito di descrivere in modo formale e verificabile i requisiti del sistema

- 1.2. Scopo: il software deve permettere di far giocare a Poker un gruppo di utenti da calcolatori diversi
- 1.3. Definizioni, acronimi e abbreviazioni
  - JVM: Java Virtual Machine, ambiente di esecuzione che permette di eseguire programmi scritti in linguaggio Java su qualsiasi piattaforma che la supporta, tramite astrazione dell'hardware sottostante
  - DB: database
  - IDE: ambiente di sviluppo del codice
  - GUI: Graphical User Interface, ambiente di controllo per il giocatore
- 1.4. Riferimenti: regolamento ufficiale di Poker
2. Descrizione generale
  - 2.1. Prospettiva del prodotto
    - Il sistema è un'applicazione client/server per giocare a Poker progettata per piattaforme computer che supportano la JVM
    - Il sistema si interfaccia con un DB per la gestione dei dati utente
    - Il sistema si dividerà in 3 moduli: client, logica di gioco, server
  - 2.2. Funzioni del prodotto
    - Gestione utenti
    - Creazione partite
    - Gestione del flusso di gioco e GUI
    - Gestione del flusso di dati
  - 2.3. Caratteristiche dell'utente
    - Admin: moderatore e gestore del sistema
    - Player: utente finale
  - 2.4. Vincoli
    - Il sistema utilizza un'architettura client/server
    - Il codice è scritto in linguaggio Java su IDE Eclipse
  - 2.5. Presupposti e dipendenze
    - Necessaria una connessione stabile ad internet
    - Il DB viene gestito con una dipendenza Maven
3. Requisiti specifici
  - 3.1. Requisiti dell'interfaccia esterna
    - 3.1.1. Interfaccia utente
      - Login e registrazione
      - Tavolo da gioco, carte e fiches
      - Pulsanti di azione
    - 3.1.2. Interfaccia hardware
      - Client: pc che supporti una JVM e periferiche di controllo
      - Server: pc che ne supporti l'esecuzione
    - 3.1.3. Interfaccia software
      - libreria log4j per gestire i log durante l'esecuzione
      - libreria per supportare l'architettura client/server
    - 3.1.4. Interfacce di comunicazione

## 3.2. Richieste funzionali

## 3.2.1. Classe Player

ID	Requisito	Descrizione
1	Registrazione e login	L'utente può creare un nuovo account oppure accedere ad uno già esistente dalla schermata di Login
2	Crea partita	L'utente può creare una partita
3	Partecipa partita	L'utente può partecipare alla partita unendosi al tavolo: <ul style="list-style-type: none"> <li>• a partita appena creata</li> <li>• dopo la fine della mano attuale</li> </ul>
4	Esci partita	L'utente può uscire dalla partita abbandonando il tavolo solo dopo la fine della mano attuale.
5	Azioni di gioco	<p>Le azioni di gioco saranno implementate sulla GUI come bottoni.</p> <p>Prima di distribuire le carte (invito):</p> <ul style="list-style-type: none"> <li>• Partecipa <ul style="list-style-type: none"> <li>○ Punta</li> </ul> </li> <li>• Non partecipare</li> </ul> <p>Appena sono state distribuite le carte (apertura):</p> <ul style="list-style-type: none"> <li>• Apri</li> <li>• Passa</li> </ul> <p>Appena uno dei giocatori apre diventa il primo del giro, quindi tutti:</p> <ul style="list-style-type: none"> <li>• Vedi</li> <li>• Punta</li> <li>• Lascia</li> </ul> <p>Quando tutti hanno deciso se lasciare o vedere (accomodo):</p> <ul style="list-style-type: none"> <li>• Cambio</li> <li>• Servito</li> </ul> <p>Quando tutti hanno gestito le proprie carte, sempre dal giocatore che aveva aperto:</p> <ul style="list-style-type: none"> <li>• Vedi</li> <li>• Punta</li> <li>• Lascia</li> </ul>
5.1	Partecipa	L'utente partecipa alla mano puntando il minimo del piatto
5.2	Non partecipare	L'utente lascia il tavolo per la mano attuale
5.3	Punta	<ul style="list-style-type: none"> <li>• L'utente può decidere la quantità da puntare inserendo il valore numerico controllando l'input con uno slider <ul style="list-style-type: none"> <li>○ puntata di partecipazione <ul style="list-style-type: none"> <li>■ tra un range di 5 e 100</li> <li>■ la sceglie il primo del giro e gli altri devono vedere o lasciare</li> </ul> </li> <li>○ puntata di apertura e post-accomodo: <ul style="list-style-type: none"> <li>■ deve essere minore o uguale alla somma dell'intero piatto e non inferiore alla puntata di invito</li> <li>■ si continua a puntare fino a quando tutti gli attivi vedono</li> </ul> </li> </ul> </li> </ul>

5.4	Apri	<ul style="list-style-type: none"> <li>• Può aprire solo il giocatore che ha almeno una coppia di Jack (ma può anche passare)</li> <li>• Quando apre punta</li> </ul>
5.5	Passa	<ul style="list-style-type: none"> <li>• Se l'utente non può aprire perchè non ha almeno una coppia di Jack allora è costretto a passare</li> <li>• Si può passare anche se si può aprire</li> <li>• Se nessuno può aprire (tutti hanno passato), il piatto rimane, si rimescolano le carte e si riparte da capo</li> </ul>
5.6	Lascia	L'utente può sempre lasciare il tavolo
5.7	Vedi	<ul style="list-style-type: none"> <li>• Nella fase di apertura solo se un giocatore ha aperto allora l'utente può vedere partecipando alla mano con una puntata che eguagli la puntata massima (il debito viene saldato)</li> <li>• Nella fase post-accomodo l'utente può vedere partecipando alla mano con una puntata che eguagli la puntata massima (il debito viene saldato)</li> </ul>
5.8	Cambio	Il giocatore cambia fino a 5 carte solo nella fase di accomodo
5.9	Servito	Il giocatore non cambia carte (solo nella fase di accomodo)

### 3.2.2. Classe Admin avrà tutti i requisiti della classe player e in più

ID	Requisito	Descrizione
		Dovrà poter gestire il contenuto del DB del sistema

### 3.2.3. Classe Poker

ID	Requisito	Descrizione
1	Credito	Il credito iniziale di ogni giocatore per la partita è definito dal tavolo ed è uguale per tutti
2	Turnazione	<ul style="list-style-type: none"> <li>• alla fine di ogni mano (solo se si arriva allo showdown) l'inizio del turno è ceduto al giocatore alla sinistra del giocatore primo nella mano attuale (il primo che si è unito al tavolo)</li> <li>• il giro è orario</li> <li>• nella fase di invito inizia il primo giocatore che si è unito al tavolo</li> <li>• nelle fasi di accomodo e puntate inizia il giocatore che ha aperto per primo</li> </ul>
3	Mazzo di carte	Deve poter gestire il mazzo di carte <ul style="list-style-type: none"> <li>• alla fine di ogni mano si mischiano le carte nel mazzo</li> </ul>
3.1	Numero di carte	In base al numero dei giocatori cambiano le carte da utilizzare secondo la formula: <ul style="list-style-type: none"> <li>• (max 52 carte)</li> </ul>
3.2	Shuffle	Deve prevedere una funzione per mescolare il mazzo di carte

3.3	Distribuire	Deve poter distribuire le carte ai giocatori in senso orario una a testa
4	Gerarchia semi	Deve poter gestire la gerarchia dei semi secondo la filosofia: <i>"come quando fuori piove"</i> , ovvero: cuori > quadri > fiori > picche
5	Puntate	Deve gestire il fatto che le puntate devono assumere un valore divisibile per 5
6	Showdown	Deve saper gestire la fase di showdown
6.1	Check vincitore	Deve poter determinare il vincitore della mano
6.1.1	Carta alta	Se con cinque carte non si riesce a formare neanche una coppia, il punto è dato dalla <b>carta più alta</b> delle cinque. A parità si prende in considerazione la seconda carta più alta, poi la terza e così via. Per mani identiche, vincerà il giocatore con la carta più alta di seme di rango superiore ( <b>non c'è parità</b> )
6.1.2	Coppia	Formata da <b>due carte dello stesso valore</b> più tre carte diverse (spaiate). Tra due coppie vince quella più alta, mentre se le due coppie presentano gli stessi valori vince quella con la terza carta più alta e così via. Se tutte le spaiate sono di uguale valore vincerà il giocatore con la coppia di seme di rango superiore ( <b>non c'è parità</b> )
6.1.3	Doppia coppia	Formata da <b>due coppie di carte dello stesso valore</b> . Tra due doppie coppie vince quella con la coppia più alta e, in caso di parità, predomina il valore della seconda coppia. Se la situazione persiste, conta la quinta carta spaiata. Se tutte le cinque carte sono di uguale valore vincerà il giocatore con la coppia più alta di seme di rango superiore ( <b>non c'è parità</b> )
6.1.4	Tris	Formato da <b>tre carte dello stesso valore</b> più due carte spaiate: tra due tris vince quello più alto. Non potendo scontrarsi nella stessa mano due tris di uguale valore ( <b>non c'è parità</b> )
6.1.5	Scala	Formata da una sequenza continua di cinque carte in cui non viene tenuto conto del seme ma solo della corretta successione, partendo da qualsiasi valore. Tra due scale vince sempre quella con il valore massimo più alto. A parità di valore massimo più alto vince quella con il seme di rango più alto ( <b>non c'è parità</b> )
6.1.6	Colore	Formato da <b>cinque carte, non in sequenza completa, dello stesso seme</b> . Vince quella con la carta di valore più alto e in caso di parità, predomina il valore della seconda carta più alta e così via. Se tutte le cinque carte sono di uguale valore, vince la mano con il seme di rango più alto ( <b>non c'è parità</b> )
6.1.7	Full	Formato da <b>un tris più una coppia</b> . Vince quello con il tris più alto ( <b>non c'è parità</b> )

6.1.8	Poker	Formato da <b>quattro carte dello stesso valore</b> . Vince la combinazione formata dalle carte più alte ( <b>non c'è parità</b> )
6.1.9	Scala colore	Formata da una <b>sequenza continua di carte dello stesso seme</b> . Vince quella con il valore massimo più alto. In caso di parità vince quella con il seme di rango più alto ( <b>non c'è parità</b> )
7	Tutti All-in	Se tutti esauriscono il proprio credito si va direttamente allo showdown

## 3.2.4. Classe Client

ID	Requisito	Descrizione
1	Invio	Deve essere in grado di gestire l'invio di informazioni al server
2	Ricezione	Deve essere in grado di gestire la ricezione di informazioni dal server

## 3.2.5. Classe Server

ID	Requisito	Descrizione
1	Invio	Deve essere in grado di gestire l'invio di informazioni ai client
2	Ricezione	Deve essere in grado di gestire la ricezione di informazioni dai client
3	Gestione gioco	Deve essere in grado di gestire il gioco
3.1		sincronizzare il flusso di gioco tra i client
4		gestire il numero di giocatori massimo ad un tavolo

## 3.2.6. Classe Database

ID	Requisito	Descrizione

## 3.2.7. Classe Carte

ID	Requisito	Descrizione
1	Seme	Ogni carta deve avere un seme tra cuori, quadri, fiori, picche
2	Ordine	Ogni carta avrà un ordine in scala
3	Valore	Ogni carta deve avere un valore assegnato: <ul style="list-style-type: none"> <li>• i numeri dal 2 al 10 hanno lo stesso valore del proprio numero</li> <li>• Jack vale 11</li> <li>• Donna vale 12</li> <li>• Re vale 13</li> <li>• Asso vale 14</li> </ul>
4	Facce	Ogni carta ha due facce: fronte e retro

### 3.2.8. Classe GUI

ID	Requisito	Descrizione
	Bottone	
	Carta	Deve poter gestire la visione delle carte
	Fronte	Il valore della carta e la sua figura
	Retro	Deve visualizzare il logo Just Poker™
	Coperte	
	Scoperte	

### 3.2.9. Classe ...

#### 3.3. Requisiti di prestazione

- Il sistema non deve presentare difetti di programmazione che portino al crash delle partite.
- Sono tollerate perdite di connessione client/server

#### 3.4. Vincoli di progettazione

- Architettura client/server
- Gestione delle dipendenze tramite Maven (come l'utilizzo della piattaforma di logging log4j)
- Versionamento e gestione della configurazione tramite GitHub
- Utilizzo di Eclipse come IDE (linguaggio Java)
- Non utilizzare servizi esterni, solo dipendenze Maven e plugin concordati

#### 3.5. Attributi del sistema software

- Come descritti nel paragrafo (1)

#### 3.6. Altri requisiti

ID	Requisito	Descrizione