

# Capitolo 2: Requisiti di Sistema

---

La definizione dettagliata dei requisiti ha rappresentato una fase cruciale nello sviluppo del chatbot AI per il supporto tecnico. Questo capitolo descrive le specifiche funzionali e non funzionali che il sistema doveva soddisfare in base alle esigenze specifiche di OnTheCloud e della piattaforma Konsolex.

## 2.1 Requisiti Funzionali

I requisiti funzionali descrivono le capacità e i servizi che il sistema deve fornire. Sono stati organizzati in categorie che riflettono le principali aree di funzionalità del chatbot.

### 2.1.1 Gestione Automatica delle Richieste di Supporto Tecnico

Il sistema deve:

- **RF-1.1:** Ricevere e interpretare richieste riguardanti problemi tecnici sulla piattaforma Konsolex
- **RF-1.2:** Analizzare il contenuto della richiesta per identificarne il tipo di problema riscontrato (server, dominio, email, container, ecc.)
- **RF-1.3:** Generare risposte accurate basate sulla knowledge base disponibile
- **RF-1.4:** Mantenere il contesto della conversazione per riferimenti successivi
- **RF-1.5:** Evitare di dare risposte prese al di fuori della knowledge base.

### 2.1.2 Interfaccia Conversazionale Multicanale

Il sistema deve:

- **RF-2.1:** Integrarsi all'interfaccia web sulla piattaforma Konsolex
- **RF-2.2:** Mantenere uno storico delle conversazioni

### 2.1.3 Sistema di Knowledge Base per Risposte Tecniche

Una knowledge base (base di conoscenza) è un archivio centralizzato di informazioni tecniche strutturate. Il sistema deve:

- **RF-3.1:** Utilizzare un supporto per archiviare e recuperare informazioni tecniche documentate
- **RF-3.2:** Organizzare la knowledge base in categorie logiche per facilitare l'accesso alle informazioni
- **RF-3.3:** Gestire meccanismi per l'aggiornamento della knowledge base quando vengono identificate nuove soluzioni o procedure

### 2.1.4 Meccanismo di Escalation verso Operatori Umani attraverso sistema di ticket

L'escalation è il processo di trasferimento di una richiesta a personale con maggiori competenze quando necessario. Un sistema di ticket è una soluzione per la gestione e tracciamento delle richieste di supporto. Il sistema deve:

- **RF-4.1:** Identificare automaticamente situazioni che richiedono intervento umano quando l'Intelligenza Artificiale (AI) non può risolvere il problema
- **RF-4.2:** Implementare un sistema di escalation che preservi il contesto della conversazione precedente

- **RF-4.3:** Notificare gli amministratori di sistema di nuove richieste di escalation che richiedono attenzione
- **RF-4.4:** Consentire il passaggio fluido tra AI e operatori umani nella stessa conversazione senza perdita di informazioni

### 2.1.5 Operazioni Tecniche Eseguibili tramite AI

Il sistema basato su Intelligenza Artificiale deve essere in grado di:

- **RF-5.1:** Aprire un ticket di supporto nel sistema di gestione delle richieste di assistenza
- **RF-5.2:** Controllare la lista dei server (computer dedicati che forniscono servizi in rete) di un utente
- **RF-5.3:** Controllare la lista dei domini (nomi identificativi di siti web, es. example.com) di un utente
- **RF-5.4:** Controllare la lista dei siti web di un utente
- **RF-5.5:** Riavviare un server quando necessario per ripristinarne il funzionamento
- **RF-5.6:** Riavviare il servizio MySQL (sistema di gestione di database relazionali open source) su un determinato server quando si verificano problemi specifici con il database

## 2.2 Requisiti Non Funzionali

I requisiti non funzionali definiscono qualità e vincoli del sistema, focalizzandosi su come il sistema deve operare piuttosto che su cosa deve fare.

### 2.2.1 Performance

- **RNF-1.1:** Il tempo di risposta del sistema deve essere inferiore a 30 secondi per garantire un'esperienza utente soddisfacente
- **RNF-1.2:** Il sistema deve implementare timeout appropriati per prevenire attese eccessive quando si verificano problemi con servizi esterni

### 2.2.2 Scalabilità

- **RNF-2.1:** L'architettura deve supportare la crescita lineare degli utenti senza richiedere modifiche architetturali significative
- **RNF-2.2:** L'aggiunta di nuove funzionalità o categorie nella knowledge base deve essere possibile senza interruzioni

### 2.2.3 Affidabilità

- **RNF-3.1:** Il sistema deve essere disponibile 24/7
- **RNF-3.2:** Nessun singolo punto di fallimento deve compromettere l'intero sistema
- **RNF-3.3:** Il sistema deve mantenere backup regolari di tutti i dati critici

### 2.2.4 Sicurezza

- **RNF-4.1:** Il sistema deve implementare controlli sulla lunghezza dell'input da parte utente e output da parte Openai, per ridurre i costi.

### 2.2.5 Usabilità

- **RNF-5.1:** Le risposte devono essere formattate in modo leggibile anche su dispositivi mobili

## 2.2.6 Manutenibilità

- **RNF-6.1:** L'architettura deve essere modulare con componenti disaccoppiati

## 2.2.7 Integrazione con Sistemi Esistenti

- **RNF-7.1:** Il sistema deve integrarsi con l'API Konsolex esistente senza richiedere modifiche a quest'ultima
- **RNF-7.2:** L'integrazione con Telegram deve supportare tutte le funzionalità necessarie
- **RNF-7.3:** L'integrazione con OpenAI deve essere resiliente ai cambiamenti nell'API