

Capitolo 3: Casi d'Uso

Questo capitolo descrive dettagliatamente i principali scenari di utilizzo del chatbot AI (sistema automatizzato di conversazione basato su Intelligenza Artificiale) per il supporto tecnico, evidenziando i flussi di interazione tra i diversi attori del sistema: utenti finali, operatori di supporto, amministratori e il sistema stesso.

A differenza dell'approccio tradizionale di sviluppo software dove i casi d'uso guidano l'implementazione, in questo progetto la documentazione formale dei casi d'uso è stata sviluppata dopo l'implementazione del sistema.

La formalizzazione a posteriori dei casi d'uso ha comunque fornito un importante strumento per validare la completezza del sistema e fornire una documentazione completa.

3.1 Interazione Utente

Lo scenario fondamentale su cui si basa l'intero sistema è l'interazione diretta tra l'utente e il chatbot AI, implementato principalmente attraverso il flusso gestito dall'endpoint `API.SEND_MESSAGE` (punto di accesso all'interfaccia di programmazione che invia i messaggi) nel file `web-server.ts` (componente del codice che gestisce le richieste web).

Caso d'uso: InterazioneBaseUtenteChatBot

Id: 1 **Breve descrizione:** L'utente invia un messaggio di testo tramite l'interfaccia di chat e riceve una risposta generata dal sistema AI. **Attori primari:** Utente **Attori secondari:** Sistema AI (Intelligenza Artificiale che elabora e risponde ai messaggi) **Precondizioni:** L'utente è registrato nel sistema con credenziali valide

Flusso Principale:

1. L'utente invia un messaggio di testo tramite l'interfaccia chat
2. Il sistema riceve il messaggio con le credenziali dell'utente
3. Il sistema verifica la validità dei parametri ricevuti
4. Il sistema recupera il record dell'utente nel database
5. Se l'utente non è presente nel database: 5.1 Include (**CreaNuovoUtente**) - procedura per registrare un nuovo utente nel sistema
6. Se l'utente ha un ticket aperto (richiesta di assistenza tecnica non ancora risolta): 6.1 Sequenza degli eventi alternativa: **Ticket aperto**
7. Il sistema salva il messaggio dell'utente nel database
8. Il sistema aggiorna lo stato dei messaggi per l'utente
9. Il sistema genera una risposta utilizzando l'AI 9.1 Include (**GenerazioneRispostaAI**) - creazione della risposta tramite AI
10. Il sistema salva la risposta dell'AI nel database
11. Il sistema invia la risposta all'utente
12. L'utente riceve la risposta generata dall'AI

Include:

1. **CreaNuovoUtente:** Quando l'utente non è ancora registrato nel sistema, viene eseguito il processo di creazione di un nuovo utente

2. **GenerazioneRispostaAI**: Processo di generazione della risposta tramite il sistema AI

Postcondizioni:

- Il messaggio dell'utente è registrato nel database
- La risposta dell'AI è registrata nel database
- L'utente visualizza la risposta generata dall'AI

Sequenza degli eventi alternativa:

1. **Parametri non validi**: Se i parametri del messaggio sono mancanti o non validi, il sistema restituisce un errore con il messaggio appropriato e il flusso termina
2. **Ticket aperto**: Se l'utente ha un ticket di supporto aperto, il messaggio viene inoltrato all'amministratore anziché all'AI.
3. **Errore di elaborazione**: Se si verifica un errore durante l'elaborazione, il sistema registra l'errore e restituisce dettagli sull'errore

Punti di estensione:

1. **Escalation a supporto umano**: Il flusso principale può essere esteso con l'apertura di un ticket quando il sistema riconosce di non poter risolvere autonomamente la richiesta. Questo punto di estensione è implementato all'interno della logica di risposta dell'AI.

Caso d'uso: CreaNuovoUtente

Id: 2 Breve descrizione: Il sistema crea un nuovo utente nel database con le informazioni necessarie, se questo non esiste già. **Attori primari:** Sistema chatBot **Attori secondari:** Sistema database, API Konsorex (piattaforma proprietaria)

Precondizioni:

- Si dispone di credenziali valide
- L'utente non è presente nel database del sistema chatBot

Flusso Principale:

1. Il sistema verifica la non esistenza dell'utente
2. Il sistema crea un nuovo thread per l'utente
3. Il sistema richiede le informazioni utente all'API della piattaforma proprietaria Konsorex tramite le credenziali
4. Il sistema estrae le informazioni dell'utente e crea un nuovo record utente nel database
5. Il sistema restituisce utente appena creato

Postcondizioni:

- Un nuovo utente è registrato nel database con tutti i dati necessari
- Un nuovo thread è associato all'utente

Sequenza degli eventi alternativa:

1. **Errore API Konsorex**: Se si verifica un errore durante la richiesta delle informazioni utente, il sistema crea comunque l'utente con le informazioni minime disponibili

2. **Errore di creazione thread OpenAI:** Se si verifica un errore durante la creazione del thread, il sistema restituisce un errore e interrompe il processo di creazione utente

Caso d'uso: InoltroMessaggioTicketAperto

Id: 3 Breve descrizione: Il sistema gestisce l'inoltro di un messaggio dell'utente all'amministratore quando è presente un ticket di supporto aperto. **Attori primari:** Utente **Attori secondari:** Amministratore, Sistema di notifica

Precondizioni:

- L'utente è registrato nel sistema
- L'utente ha un ticket di supporto aperto
- L'utente ha inviato un nuovo messaggio

Flusso Principale:

1. Il sistema identifica la presenza di un ticket aperto associato all'utente
2. Il sistema recupera l'ID del ticket più recente dal database
3. Il sistema salva il messaggio dell'utente nel database associandolo al ticket
4. Il sistema inoltra il messaggio agli amministratori di supporto
5. Il sistema aggiorna lo stato di risposta del cliente, resettandolo
6. Il sistema invia una notifica di aggiornamento al pannello amministrativo
7. Il sistema aggiorna lo stato dei messaggi per l'amministratore

Postcondizioni:

- Il messaggio dell'utente è registrato nel database associato al ticket
- Gli amministratori di supporto hanno ricevuto il messaggio
- Lo stato di notifica nel pannello amministrativo è aggiornato

Sequenza degli eventi alternativa:

1. **Errore di inoltro:** Se si verifica un errore nell'inoltro del messaggio all'amministratore, il sistema registra comunque il messaggio nel database e tenta nuovamente l'invio in seguito

Caso d'uso: GenerazioneRispostaAI

Id: 4 Breve descrizione: Il sistema utilizza l'API OpenAI di chatGPT per generare una risposta contestuale al messaggio dell'utente. **Attori primari:** Sistema AI **Attori secondari:** API OpenAI

Precondizioni:

- L'utente è stato identificato nel sistema
- Il messaggio dell'utente è stato salvato nel database
- L'utente ha un thread OpenAI valido

Flusso Principale:

1. Il sistema aggiunge il messaggio dell'utente al thread OpenAI associato
2. Il sistema crea una nuova esecuzione (run) nel thread con l'assistant associato all'utente

3. Il sistema fornisce all'AI gli strumenti disponibili per l'esecuzione di funzioni specializzate
4. Il sistema attende che l'elaborazione AI sia completata, monitorando lo stato dell'esecuzione
5. Se l'AI richiede l'esecuzione di funzioni specifiche: 5.1 extend (**EsecuzioneFunzioniSpecializzate**)
6. Il sistema recupera la risposta generata dall'AI
7. Il sistema controlla la pertinenza della risposta e assegna un punteggio di rilevanza
8. Il sistema formatta la risposta per la visualizzazione all'utente

Postcondizioni:

- Una risposta contestuale è stata generata dall'AI
- La risposta è stata formattata per la visualizzazione
- Un punteggio di rilevanza è stato assegnato alla risposta

Sequenza degli eventi alternativa:

1. **Risposta non valida:** Se la risposta generata non è valida, il sistema restituisce un messaggio di errore predefinito
2. **Timeout:** Se l'elaborazione richiede troppo tempo, il sistema crea un nuovo thread e restituisce un messaggio di errore
3. **Errore API:** Se si verifica un errore con l'API OpenAI, il sistema gestisce l'eccezione e restituisce un messaggio appropriato

Punti di estensione

1. **Esecuzione di funzioni specializzate:** Nella fase di elaborazione AI, il sistema può estendere il flusso base identificando la necessità di eseguire funzioni specifiche (tool calls) come il riavvio server o il controllo dello stato dei servizi. Questo è implementato nella funzione `storeMessageAndCreateReply` che processa la risposta dell'AI e identifica eventuali richieste di esecuzione funzioni.

Caso d'uso: EsecuzioneFunzioniSpecializzate

Id: 5 Breve descrizione: Il sistema esegue operazioni tecniche specifiche sui server o domini in base alla richiesta dell'utente interpretata dall'AI. **Attori primari:** Sistema AI **Attori secondari:** API Konsolex, Sistemi server

Precondizioni:

- L'AI ha identificato una richiesta di esecuzione di funzione nel messaggio dell'utente

Flusso Principale:

1. L'AI richiede l'esecuzione di una funzione specifica con i relativi parametri
2. Il sistema identifica la funzione richiesta tra quelle disponibili
3. Il sistema raccoglie e valida i parametri necessari per l'esecuzione
4. Il sistema invia la richiesta all'API Konsolex con i parametri appropriati
5. Il sistema riceve e interpreta la risposta dall'API
6. Il sistema formatta la risposta in un formato comprensibile per l'utente
7. Il sistema fornisce il risultato all'AI per completare la risposta all'utente

Postcondizioni:

- L'operazione tecnica richiesta è stata eseguita o è stato fornito un motivo del fallimento
- Il risultato dell'operazione è stato formattato e fornito all'AI

Sequenza degli eventi alternativa:

1. **Parametri invalidi:** Se i parametri forniti non sono validi o sufficienti, il sistema richiede all'utente i dati corretti
2. **Errore di esecuzione:** Se l'operazione fallisce durante l'esecuzione, il sistema cattura l'errore e fornisce informazioni sul fallimento

Caso d'uso: AperturaTicketSupporto

Id: 6 Breve descrizione: L'AI richiede assistenza umana, creando un ticket di supporto che viene inoltrato agli amministratori. **Attori primari:** Utente, Sistema AI **Attori secondari:** Amministratore

Precondizioni:

- L'utente è registrato nel sistema
- L'utente non ha già un ticket di supporto aperto

Flusso Principale:

1. L'AI identifica una richiesta che non può gestire autonomamente
2. Il sistema crea un nuovo ticket di supporto associato all'utente
3. Il sistema recupera le informazioni dell'utente (nome, ID) per il ticket
4. Il sistema compone un messaggio di ticket con le informazioni dell'utente e la richiesta originale
5. Il sistema aggiorna lo stato del messaggio associato come ticket
6. Il sistema inoltra il ticket agli amministratori tramite Telegram e la piattaforma web
7. Il sistema notifica l'utente che la sua richiesta è stata inoltrata a un operatore umano

Postcondizioni:

- Un ticket di supporto è stato creato nel database
- Gli amministratori hanno ricevuto una notifica con il ticket
- Lo stato dell'utente è stato aggiornato per indicare un ticket aperto

Sequenza degli eventi alternativa:

1. **Errore di notifica:** Se la notifica agli amministratori fallisce, il sistema riprova o registra l'errore

Caso d'uso: RispostaAmministratoreATicket

Id: 7 Breve descrizione: Un amministratore risponde a un ticket di supporto aperto e la risposta viene inoltrata all'utente. **Attori primari:** Amministratore **Attori secondari:** Utente

Precondizioni:

- Esiste un ticket di supporto aperto per l'utente
- L'amministratore ha selezionato il ticket a cui rispondere

Flusso Principale:

1. L'amministratore compone una risposta al ticket
2. Il sistema riceve la risposta con l'ID utente associato al ticket
3. Il sistema verifica la validità della richiesta e l'esistenza dell'utente
4. Il sistema recupera il ticket più recente associato all'utente
5. Il sistema salva la risposta dell'amministratore nel database, associandola al ticket
6. Il sistema inoltra la risposta all'utente
7. L'amministratore può decidere di:
 - 7.1 ripetere l'operazione e mandare un nuovo messaggio
 - 7.2 attendere una risposta
 - 7.3 extension (**ChiusuraTicketSupporto**)

Postcondizioni:

- La risposta dell'amministratore è registrata nel database
- L'utente ha ricevuto la risposta
- Lo stato del ticket è aggiornato

Sequenza degli eventi alternativa:

1. **Utente non trovato:** Se l'utente associato al ticket non viene trovato, il sistema restituisce un errore
2. **Errore di inoltro:** Se l'inoltro della risposta all'utente fallisce, il sistema registra l'errore

Caso d'uso: ChiusuraTicketSupporto

Id: 8 Breve descrizione: Un amministratore chiude un ticket di supporto dopo che la richiesta dell'utente è stata risolta. **Attori primari:** Amministratore **Attori secondari:** Utente

Precondizioni:

- Esiste un ticket di supporto aperto per l'utente
- L'amministratore ha determinato che il problema è stato risolto

Flusso Principale:

1. L'amministratore richiede la chiusura del ticket specificando l'ID utente
2. Il sistema verifica che il ticket esista per l'utente specificato
3. Il sistema aggiorna lo stato del ticket come "chiuso" nel database
4. Il sistema aggiorna lo stato dell'utente per indicare che non ha più ticket aperti
5. Il sistema notifica l'amministratore del successo dell'operazione

Postcondizioni:

- Il ticket è marcato come chiuso nel database
- L'utente non ha più ticket aperti
- Le future interazioni dell'utente saranno nuovamente gestite dall'AI

Sequenza degli eventi alternativa:

1. **Ticket non trovato:** Se non viene trovato alcun ticket aperto per l'utente specificato, il sistema restituisce un errore
2. **Errore di aggiornamento:** Se si verifica un errore durante l'aggiornamento dello stato del ticket, il sistema registra l'errore