



SkyNet

Analisi dei requisiti

Bot4Me

skynet.swe@gmail.com

22 Aprile 2022

Redattori: Anna Cisotto Bertocco, Nicholas Pilotto, Alberto Matterazzo, Davide Dinato

Verificatori: Davide Sut

Responsabile: Kaltrina Collaku

Destinatari: Prof. Tullio Vardanega, Prof. Riccardo Cardin

Uso: **Esterno**

Stato: Approvato

Versione: **1.0.10**

Registro delle Modifiche

Versione	Autore	Data	Ruolo	Descrizione
1.0.10	Davide Sut	22-06-2022	Responsabile	Miglioramento sezioni §3.5, §3.6, stesura §3.17
1.0.9	Davide Dinato	21-06-2022	Verificatore	Aggiornato tracciamento requisiti, controllo ortografia
1.0.8	Alberto Matterazzo	21-06-2022	Analista	Stesura §3.16, §3.16.1, §3.16.2
1.0.7	Anna Cisotto Bertocco	20-06-2022	Analista	Stesura §3.12, §3.15, §3.17
1.0.6	Nicholas Pilotto	18-06-2022	Analista	Migliorie §3.8, §3.8.1, §3.8.2, §3.8.3, §3.8.4, §3.8.5, §3.8.6, §3.8.7, §3.8.8
1.0.5	Anna Cisotto Bertocco	19-06-2022	Analista	Modifica §2.4.2, §2.2, §3.2, §3.3 e §3.4, stesura §3.10, §3.11 e §3.12
1.0.4	Davide Sut	18-06-2022	Analista	Estensione §3.4, §3.5 e §3.6 e stesura relative sotto sezioni
1.0.3	Nicholas Pilotto	18-06-2022	Verificatore	Miglioria §3.9
1.0.2	Davide Dinato	18-06-2022	Amministratore	Migliorie vincoli R2V1, R2V2, R2V6, R2V7 e casi d'uso, aggiunti requisiti prestazionali, stesura UC13
1.0.1	Nicholas Pilotto	18-06-2022	Analista	Scrittura §3.8, §3.8.1, §3.8.2, §3.8.3, §3.8.4, §3.8.5, §3.8.6, §3.8.7, §3.8.8
1.0.0	Kaltrina Collaku	13-06-2022	Responsabile	Approvazione documento
0.2.0	Davide Sut	12-06-2022	Verificatore	Verifica documento
0.1.4	Kaltrina Collaku	4-06-2022	Analista	Modifica §4.1, §5.1, §5.2
0.1.3	Nicholas Pilotto	03-06-2022	Analista	Aggiunta vocaboli al Documento Glossario
0.1.2	Nicholas Pilotto	25-05-2022	Analista	Aggiunta §3.2, §3.3, §3.4, §3.5, §3.6, §3.7, §3.8, §3.9
0.1.1	Davide Sut	23-04-2022	Analista	Stesura §2.5
0.1.0	Davide Sut	22-04-2022	Verificatore	Verifica documento
0.0.6	Anna Cisotto Bertocco	22-04-2022	Analista	Stesura §2.3 e §2.4
0.0.5	Davide Dinato	21-04-2022	Analista	Stesura §4.2
0.0.4	Alberto Matterazzo	20-04-2022	Analista	Stesura da §3.1 a §3.6
0.0.3	Nicholas Pilotto	20-04-2022	Analista	Stesura §4.1, §4.3 e §4.4



Versione	Autore	Data	Ruolo	Descrizione
0.0.2	Anna Cisotto Bertocco	19-04-2022	Analista	Stesura §2.1 e §2.2
0.0.1	Davide Sut	28-03-2022	Amministratore	Creazione struttura documento



Indice

1	Introduzione	1
1.1	Scopo del documento	1
1.2	Glossario	1
1.3	Riferimenti	1
1.3.1	Normativi	1
1.3.2	Informativi	1
2	Descrizione del prodotto	2
2.1	Scopo del prodotto	2
2.2	Analisi della struttura	2
2.3	Utenti target	2
2.4	Attori	2
2.4.1	Attori principali	3
2.4.2	Attori secondari	3
2.5	Vincoli generali	3
3	Casi d'uso	4
3.1	Scopo	4
3.2	UC1	4
3.2.1	UC1.1	4
3.2.2	UC1.2	5
3.3	UC2	5
3.3.1	UC2.1	6
3.3.2	UC2.2	6
3.3.3	UC2.3	6
3.3.4	UC2.4	7
3.3.5	UC2.5	7
3.4	UC3	7
3.4.1	UC3.1	8
3.4.2	UC3.2	8
3.4.3	UC3.3	8
3.4.4	UC3.4	9
3.4.5	UC3.5	9
3.4.6	UC3.6	9
3.5	UC4	10
3.5.1	UC4.1	10
3.5.2	UC4.2	11
3.5.3	UC4.3	11
3.5.4	UC4.4	11
3.5.5	UC4.5	11
3.5.6	UC4.6	12
3.5.7	UC4.7	12
3.5.8	UC4.8	12
3.6	UC5	12
3.6.1	UC5.1	13
3.6.2	UC5.2	13
3.7	UC6	13
3.7.1	UC6.1	14
3.7.2	UC6.2	14
3.7.3	UC6.3	15
3.7.4	UC6.4	15
3.8	UC7	15
3.8.1	UC7.1	16
3.8.2	UC7.2	16



3.8.3	UC7.3	16
3.8.4	UC7.4	17
3.8.5	UC7.5	17
3.8.6	UC7.6	17
3.8.7	UC7.7	18
3.8.8	UC7.8	18
3.9	UC8	18
3.10	UC9	19
3.11	UC10	19
3.12	UC11	19
3.13	UC12	19
3.14	UC13	20
3.14.1	UC13.1	20
3.14.2	UC13.2	20
3.15	UC14	21
3.16	UC15	21
3.16.1	UC15.1	21
3.16.2	UC15.2	22
3.17	UC16	22
3.18	UC17	22
4	Requisiti	23
4.1	Requisiti funzionali	23
4.2	Requisiti prestazionali	27
4.3	Requisiti qualitativi	27
4.4	Requisiti di vincolo	28
5	Tracciamento requisiti	30
5.1	Requisito - Fonte	30
5.2	Fonte - Requisito	34



Elenco delle tabelle

1	Requisiti funzionali	27
2	Requisiti prestazionali	27
3	Requisiti qualitativi	28
4	Requisiti di vincolo	29
5	Requisito - Fonte	33
6	Fonte - Requisito	34



Elenco delle figure

1	Attori	3
---	------------------	---

1 Introduzione

1.1 Scopo del documento

Il seguente documento ha come obiettivo quello di trattare in modo completo ed esaustivo la formulazione dei casi d'uso e dei requisiti evidenziati dall'analisi del capitolato d'appalto *Bot4Me* (C1) secondo le esigenze del proponente *Imola Informatica*.

1.2 Glossario

Per evitare incomprensioni e ambiguità durante la lettura del documento, vengono utilizzati due simboli a pedice di alcuni termini, con le seguenti funzioni:

- G per indicare i termini la cui definizione si trova nel Glossario v1.0.0 $_D$
- D per indicare il nome di un documento esterno

1.3 Riferimenti

1.3.1 Normativi

- Norme di Progetto v0.1.0 $_D$
- Capitolato: <https://www.math.unipd.it/tullio/IS-1/2021/Progetto/C1.pdf>

1.3.2 Informativi

- Slide del corso - lezione T7: Analisi dei requisiti:
<https://www.math.unipd.it/tullio/IS-1/2021/Dispense/T07.pdf>
- Slide del corso - lezione P4: Diagrammi dei casi d'uso:
<https://www.math.unipd.it/rcardin/swea/2022/Diagrammi%20Use%20Case.pdf>

2 Descrizione del prodotto

2.1 Scopo del prodotto

Lo scopo del prodotto è quello di semplificare le attività aziendali di routine mediante l'utilizzo di un *ChatBot_G*, rendendo possibile un'interazione sia testuale che vocale con i dipendenti di Imola Informatica. L'applicativo finale sarà una *Web App_G* accessibile sia da dispositivi *mobile_G*, quali *smartphone* e *tablet*, sia da dispositivi *desktop_G* tramite *browser_G*.

2.2 Analisi della struttura

Le componenti principali in cui è strutturato il prodotto sono le seguenti:

- **CHATBOT-FE_G**: componente per il *front-end_G* dell'applicazione, che si occupa di creare l'interfaccia in stile app di messaggistica (es. *Telegram_G*, *Discord_G*) con cui interagisce l'utente. Tramite l'interfaccia principale della *chat* l'utente potrà svolgere le seguenti azioni:
 - inviare un messaggio (testuale o vocale) al *ChatBot_G*
 - visualizzare il flusso di messaggi inviati a partire dall'inizio della connessione
 - fare una richiesta specifica al *ChatBot_G*
- **CHATBOT-BE_G**: componente per il *back-end_G* dell'applicazione, che si occupa di gestire le richieste degli utenti attraverso i servizi *REST_G* forniti da Imola Informatica. Questo componente si occupa quindi di:
 - interpretare i messaggi inviati dall'utente
 - richiedere eventuali informazioni mancanti per rispondere alla richiesta
 - rispondere alla richiesta

2.3 Utenti target

Date le caratteristiche del prodotto richieste dal capitolato e da Imola Informatica, gli utenti che potranno usufruire dell'applicativo saranno esclusivamente i dipendenti dell'azienda, ovvero tutti gli utenti che dispongono di una email del tipo *utente@imolainformatica.it*.

2.4 Attori

Gli attori_G. individuati dal gruppo a seguito dell'analisi del capitolato e degli incontri con Imola Informatica sono i seguenti:



Figura 1: Attori

2.4.1 Attori principali

- **Utente non autenticato:** utente che non ha ancora eseguito l'accesso tramite email aziendale e quindi non ancora identificato come dipendente dell'azienda; non ha accesso alle varie funzionalità dell'applicativo.
- **Utente autenticato:** utente che ha eseguito l'accesso tramite email aziendale e che quindi può usufruire di tutti i servizi forniti dall'applicativo.

Non viene fatta alcuna distinzione tra gli utenti autenticati, i quali hanno quindi tutti a disposizione l'intera gamma di funzionalità dell'applicativo.

2.4.2 Attori secondari

- **Piattaforma *Redmine_G*:** sistema software per il tracciamento delle segnalazioni di bug tramite *ticket_G* che gli attori_G potranno creare tramite il *ChatBot_G*.

2.5 Vincoli generali

Per poter utilizzare correttamente l'applicativo è necessario rispettare i seguenti vincoli:

- Possedere una connessione ad Internet attiva e che non limiti le connessioni da e verso i servizi utili all'applicativo.
- Avere a disposizione un dispositivo *mobile_G* o *desktop_G* provvisto di *browser web_G* per poter accedere all'applicativo.
- Possedere un metodo di input valido, tastiera o microfono, attraverso il quale è possibile interagire con la *Web App_G*.

3 Casi d'uso

3.1 Scopo

Lo scopo di questa sezione è elencare e descrivere tutti i casi d'uso_G individuati dal gruppo *SkyNet*, in riferimento alle funzionalità dell'applicativo.

Ogni caso d'uso_G rappresenta un insieme di scenari che hanno in comune uno scopo finale per un utente generico del sistema, chiamato *attore_G*. I casi d'uso possono essere descritti tramite *diagrammi dei casi d'uso_G*, possono estendersi in più sotto casi e possiedono una precondizione seguita da una post condizione. Ogni caso d'uso è definito dal suo nome e tramite la seguente struttura:

- **Attori:** indica gli attori_G principali e secondari del caso d'uso;
- **Descrizione:** viene riportata una breve descrizione del caso d'uso;
- **Precondizione:** specifica le condizioni che sono identificate come vere prima del verificarsi degli eventi del caso d'uso;
- **Post condizione:** specifica le condizioni che sono identificate come vere dopo il verificarsi degli eventi del caso d'uso;
- **Scenario principale:** rappresenta il flusso degli eventi;
- **Estensioni:** specifica i casi d'uso che estendono questo caso d'uso;

3.2 UC1

Login_G

- **Attori:** utente non autenticato.
- **Descrizione:** per usufruire delle varie funzionalità dell'applicativo l'utente deve essere autenticato.
- **Precondizione:** l'utente non è autenticato e vuole autenticarsi nel sistema tramite inserimento dell'API Key_G univoca fornita da Imola Informatica.
- **Post condizione:** l'utente si è autenticato con successo all'interno del sistema tramite l'inserimento dell'API Key_G e quindi ha accesso a tutti i servizi forniti dall'applicativo.
- **Scenario principale:** l'utente inserisce l'API Key_G personale fornita da Imola Informatica per effettuare il login_G al ChatBot_G.
- **Estensioni:**
 - il ChatBot_G non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).
 - l'utente inserisce una API Key_G in un formato non valido (UC1.1).
 - l'utente inserisce una API Key_G errata o non esistente (UC1.2).

3.2.1 UC1.1

API Key_G non valida.

- **Attori:** utente non autenticato.
- **Descrizione:** l'API Key_G inserita dall'utente è espressa in un formato non valido.
- **Precondizione:** l'utente non è autenticato e ha inserito un'API Key_G in un formato non valido.
- **Post condizione:** il ChatBot_G, dopo aver controllato i dati inseriti, notifica l'utente dell'errore e richiede di inserire nuovamente l'API Key_G corretta.

- **Scenario principale:**

- Il *ChatBot_G* chiede "L'API Key che hai inserito risulta espressa in un formato non valido. Per effettuare il login inserisci nuovamente l'API Key".
- L'utente inserisce nuovamente l'API Key_G tramite la *chat_G*.

3.2.2 UC1.2

API Key_G errata.

- **Attori:** utente non autenticato.
- **Descrizione:** l'API Key_G inserita dall'utente è errata o non esistente.
- **Precondizione:** l'utente non è autenticato e ha inserito un'API Key_G errata o non esistente.
- **Post condizione:** il *ChatBot_G* notifica l'utente che l'API Key_G inserita è errata.
- **Scenario principale:** Il *ChatBot_G* manda il messaggio "L'API Key che hai inserito risulta errata. Per effettuare il login inserisci API Key valida".

3.3 UC2

Tracciamento della presenza in sede in EMT_G

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente dell'azienda vuole tracciare la propria presenza in una sede attraverso l'applicativo aziendale EMT_G.
- **Precondizione:** un dipendente deve ancora tracciare la propria presenza attraverso l'applicativo EMT_G.
- **Post condizione:** il dipendente ha segnalato al *ChatBot_G* la presenza in sede con successo.
- **Scenario principale:** se il dipendente non si è ancora autenticato nell'applicativo deve effettuare la procedura di *login_G* (UC1). A questo punto il dipendente invia un messaggio del tipo:
 - "Sono arrivato nella sede di XY" o "Voglio fare il check-in per la sede di XY" (UC2.1)
 - "Sto andando via" o "Voglio fare il check-out" (UC2.2)

Il sistema riceve il messaggio e, dopo aver verificato che i dati inseriti non siano errati, richiede delle informazioni mancanti qualora non fossero presenti nel messaggio originale. Una volta inserite tali informazioni, il sistema registra la presenza del dipendente utilizzando le *API_G* dell'applicativo EMT_G e conferma l'esecuzione dell'operazione.

Le informazioni che il *ChatBot_G* può richiedere se non presenti nel messaggio sono:

- La sede in cui si intende registrare la presenza.
- **Estensioni:**
 - il *ChatBot_G* non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).
 - la sede per cui si desidera registrare la presenza risulta inesistente (UC10).
 - l'utente inserisce una sede in un formato non valido (UC2.3).

3.3.1 UC2.1

Check-in_G per una sede

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente dell'azienda vuole effettuare il *check-in_G* per una sede aziendale.
- **Precondizione:** un dipendente ha espresso la volontà di effettuare il *check-in* in una sede al *ChatBot_G*.
- **Post condizione:** il dipendente ha inserito la sede in cui fare il *check-in*.
- **Scenario principale:** l'utente invia un messaggio del tipo "*Voglio fare il check-in per la sede di XY*". Il *ChatBot_G* notifica l'utente dell'avvenuta registrazione della presenza nella sede.
- **Estensioni:** *Check-in_G* già effettuato per una sede (UC2.4)

3.3.2 UC2.2

Check-out_G per una sede

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente dell'azienda vuole effettuare il *check-out_G* per la sede in cui si trova.
- **Precondizione:** un dipendente ha fatto il *check-in_G* per una sede e vuole effettuare il *check-out* da quella sede.
- **Post condizione:** il *ChatBot_G* notifica l'utente dell'avvenuto *check-out_G* dalla sede in cui è stata registrata la presenza.
- **Scenario principale:** l'utente invia un messaggio del tipo "*Voglio fare il check-out*". Il *ChatBot_G* chiede all'utente conferma della sede che si desidera lasciare con il messaggio "*Vuoi fare il check-out per la sede XY?*" e poi notifica l'utente dell'avvenuto *check-out_G* dalla sede.
- **Estensioni:** *Check-out_G* già effettuato per una sede (UC2.5)

3.3.3 UC2.3

Sede non valida.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente ha inserito la sede presso cui registrare la presenza in un formato non valido.
- **Precondizione:** l'utente ha inserito una sede in un formato non valido.
- **Post condizione:** il *ChatBot_G*, dopo aver controllato i dati inseriti, notifica l'utente dell'errore e richiede di inserire nuovamente una sede in cui registrare la presenza.
- **Scenario principale:**
 - Il *ChatBot_G* chiede "*La sede per cui desideri registrare la presenza risulta espressa in un formato non valido. Inserisci nuovamente la sede*".
 - L'utente inserisce nuovamente la sede tramite la *chat_G*.

3.3.4 UC2.4

Check-in_G già effettuato per una sede.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente richiede di fare il *check-in_G* in una sede ma ha già registrato la presenza per una sede.
- **Precondizione:** l'utente richiede di fare il *check-in_G* presso una sede ma ha già registrato la presenza per una sede.
- **Post condizione:** il *ChatBot_G* notifica l'utente del fatto che sia già stata registrata una presenza in una sede.
- **Scenario principale:** Il *ChatBot_G* manda il messaggio "*Hai già effettuato il check-in*".

3.3.5 UC2.5

Check-out_G già effettuato per una sede.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente richiede di fare il *check-out_G* in una sede ma ha già fatto il *check-out_G* per una sede.
- **Precondizione:** l'utente richiede di fare il *check-out_G* presso una sede ma ha già fatto il *check-out_G* per una sede.
- **Post condizione:** il *ChatBot_G* notifica l'utente del fatto che sia già stato richiesto un *check-out_G* per una sede.
- **Scenario principale:** Il *ChatBot_G* manda il messaggio "*Hai già effettuato il check-out per questa sede*".

3.4 UC3

Inserimento di una nuova attività in EMT_G

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente vuole inserire una nuova attività giornaliera_G nell'applicativo aziendale EMT_G. In questo modo il dipendente attraverso la registrazione delle attività svolte può compilare il diario giornaliero per raggiungere le ore lavorative giornaliere necessarie.
- **Precondizione:** un dipendente deve ancora inserire l'attività svolta nell'applicativo.
- **Post condizione:** il dipendente ha registrato con successo l'attività nell'applicativo EMT_G.
- **Scenario principale:** se il dipendente non si è ancora autenticato nell'applicativo deve effettuare la procedura di *login_G* (UC1). A questo punto il dipendente invia un messaggio del tipo "*Ho lavorato N ore sul progetto XY*". Il sistema riceve il messaggio e richiede delle informazioni mancanti qualora non fossero presenti nel messaggio originale. Una volta inserite tali informazioni, il sistema registra l'attività svolta utilizzando le API_G dell'applicativo EMT_G, conferma il corretto inserimento dell'attività a calendario e restituisce le eventuali ore mancanti a raggiungere le ore lavorative giornaliere (UC12) Le informazioni richieste sono:
 - il codice del progetto per cui si vuole consuntivare l'attività (UC3.1).
 - il numero di ore da consuntivare (UC3.2)
 - la descrizione dell'attività svolta (UC3.3)
- **Estensioni:**
 - il *ChatBot_G* non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).

3.4.1 UC3.1

Inserimento codice progetto

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente chiede di consuntivare un'attività per un progetto identificato da un codice.
- **Precondizione:** l'utente vuole inserire il codice del progetto per cui desidera consuntivare l'attività.
- **Post condizione:** l'utente ha inserito con successo il codice del progetto per cui desidera consuntivare l'attività.
- **Scenario principale:**
 - Il *ChatBot_G* chiede "Per quale progetto vuoi consuntivare l'attività?".
 - l'utente inserisce tramite *chat_G* il codice del progetto.
- **Estensioni:**
 - l'utente inserisce un codice di un progetto in un formato non valido (UC3.4)
 - il progetto corrispondente al codice inserito non esiste (UC11)

3.4.2 UC3.2

Inserimento numero di ore da consuntivare

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente chiede di consuntivare un numero di ore per l'attività.
- **Precondizione:** l'utente vuole inserire il numero di ore da consuntivare.
- **Post condizione:** l'utente ha inserito con successo il numero di ore da consuntivare.
- **Scenario principale:**
 - Il *ChatBot_G* chiede "Quante ore vuoi consuntivare?".
 - l'utente inserisce tramite *chat_G* il numero di ore che desidera consuntivare.
- **Estensioni:** l'utente inserisce un numero di ore in un formato non valido (UC3.5)

3.4.3 UC3.3

Inserimento descrizione dell'attività

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente chiede di inserire una descrizione per l'attività.
- **Precondizione:** l'utente vuole inserire una descrizione per l'attività.
- **Post condizione:** l'utente ha inserito con successo una descrizione per l'attività.
- **Scenario principale:**
 - Il *ChatBot_G* chiede "Inserisci una breve descrizione dell'attività".
 - l'utente inserisce tramite *chat_G* la descrizione per l'attività.
- **Estensioni:** l'utente inserisce una descrizione in un formato non valido (UC3.6)

3.4.4 UC3.4

Codice progetto non valido

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente inserisce un codice per il progetto in un formato non valido.
- **Precondizione:** l'utente ha inserito un codice per il progetto in un formato non valido.
- **Post condizione:** il *ChatBot_G* notifica l'utente che il codice del progetto non è valido e richiede di inserire nuovamente il codice.
- **Scenario principale:**
 - Il *ChatBot_G* chiede *"Il codice progetto che hai inserito risulta espresso in un formato non valido. Inserisci nuovamente il codice del progetto"*.
 - l'utente inserisce tramite *chat_G* il codice del progetto.

3.4.5 UC3.5

Numero di ore non valido

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente inserisce le ore da consuntivare in un formato non valido.
- **Precondizione:** l'utente ha inserito le ore da consuntivare in un formato non valido.
- **Post condizione:** il *ChatBot_G* notifica l'utente che le ore da consuntivare sono espresse in un formato non valido e richiede di inserire nuovamente il numero di ore.
- **Scenario principale:**
 - Il *ChatBot_G* chiede *"Il numero di ore lavorate è in un formato non valido. Inserisci nuovamente il numero di ore da consuntivare"*.
 - l'utente inserisce tramite *chat_G* il numero di ore da consuntivare.

3.4.6 UC3.6

Descrizione dell'attività non valida

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente inserisce la descrizione dell'attività in un formato non valido.
- **Precondizione:** l'utente ha inserito la descrizione dell'attività in un formato non valido.
- **Post condizione:** il *ChatBot_G* notifica l'utente che la descrizione dell'attività è espressa in un formato non valido e richiede di inserire nuovamente la descrizione.
- **Scenario principale:**
 - Il *ChatBot_G* chiede *"La descrizione dell'attività è in un formato non valido. Inserisci nuovamente una breve descrizione per l'attività"*.
 - l'utente inserisce tramite *chat_G* la descrizione dell'attività.

3.5 UC4

Inserimento di una nuova riunione su un contesto applicativo esterno $_G$

- **Attori:** utente autenticato tramite API key.
- **Descrizione:** un dipendente vuole creare una riunione su un applicativo per videoconferenze con l'orario e i partecipanti desiderati.
- **Precondizione:** un dipendente non ha ancora creato la riunione desiderata sull'applicativo per videoconferenze.
- **Post condizione:** il dipendente è riuscito ad inserire con successo e a programmare la riunione nell'applicativo.
- **Scenario principale:** se il dipendente non si è ancora autenticato nell'applicativo deve effettuare la procedura di $login_G$ (UC1). A questo punto il dipendente invia un messaggio del tipo "*Crea una riunione su X per la data Y alle ore Z con nome1.cognome1@imolainformatica.it, nome2.cognome2@imolainformatica.it*". Il sistema verifica che non ci siano sovrapposte altre riunioni già pianificate. In tal caso il sistema utilizza le API_G dello strumento per inserire la riunione e qualora fosse necessario richiede delle informazioni aggiuntive. Una volta inserite tali informazioni, il sistema conferma la corretta programmazione della riunione e restituisce l'elenco delle riunioni giornaliere del dipendente.
Le informazioni richieste sono le seguenti:
 - la piattaforma sulla quale effettuare la riunione. (UC4.1)
 - la data della riunione. (UC4.3)
 - l'orario della riunione. (UC4.5)
 - i membri che partecipano alla riunione. (UC4.7)
- **Estensioni:**
 - il $ChatBot_G$ non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).
 - visualizzazione riunioni giornaliere (UC17).

3.5.1 UC4.1

Inserimento della piattaforma sulla quale effettuare la riunione.

- **Attori:** utente autenticato tramite API key $_G$.
- **Descrizione:** un dipendente comunica al chatbot la piattaforma sulla quale vuole creare una riunione.
- **Precondizione:** il dipendente ha espresso la volontà di voler creare una nuova riunione al chatbot.
- **Post condizione:** il dipendente ha comunicato al chatbot la piattaforma sulla quale vuole creare una riunione.
- **Scenario principale:** il chatbot chiede all'utente il nome della piattaforma sulla quale vuole creare la riunione con un messaggio del tipo "*Che piattaforma vuoi usare?*". Il dipendente risponde con il nome della piattaforma.
- **Estensione:** il dipendente risponde con un nome di piattaforma non valido. (UC4.2)

3.5.2 UC4.2

Piattaforma sulla quale effettuare la riunione non valida.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** la piattaforma comunicata al chatbot è non valida.
- **Precondizione:** un dipendente ha comunicato al chatbot la piattaforma sulla quale vuole creare una riunione in un formato non valido o essa non è presente tra le piattaforme approvate dall'azienda.
- **Post condizione:** il chatbot ha comunicato al dipendente che la piattaforma non è valida.
- **Scenario principale:** il chatbot comunica all'utente che la piattaforma inserita non è valida tramite un messaggio testuale del tipo *"La piattaforma inserita non è valida"*.

3.5.3 UC4.3

Inserimento della data della riunione.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente comunica al chatbot la data nella quale verrà effettuata la riunione.
- **Precondizione:** il dipendente ha espresso la volontà di voler creare una nuova riunione al chatbot.
- **Post condizione:** il dipendente ha comunicato al chatbot la data della riunione.
- **Scenario principale:** il chatbot chiede all'utente la data della riunione con un messaggio del tipo *"In quale data vuoi fissare la riunione?"*.
Il dipendente risponde comunicando al chatbot una data.
- **Estensione:** il dipendente risponde con una data non valida. (UC4.4)

3.5.4 UC4.4

Data della riunione non valida.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** la data comunicata al chatbot è non valida.
- **Precondizione:** il dipendente ha comunicato al chatbot la data della riunione in un formato non valido o essa è festiva o non disponibile.
- **Post condizione:** il chatbot ha comunicato al dipendente che la data inserita non è valida.
- **Scenario principale:** il chatbot comunica all'utente che la data inserita non è valida tramite un messaggio testuale del tipo *"La data inserita non è valida"*.

3.5.5 UC4.5

Inserimento dell'orario della riunione.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente comunica al chatbot l'orario nel quale verrà effettuata la riunione.
- **Precondizione:** il dipendente ha espresso la volontà di voler creare una nuova riunione al chatbot.
- **Post condizione:** il dipendente ha comunicato al chatbot l'orario della riunione.
- **Scenario principale:** il chatbot chiede all'utente l'orario della riunione con un messaggio del tipo *"A che ora vuoi effettuare la riunione?"*.
Il dipendente risponde comunicando al chatbot un orario.
- **Estensione:** il dipendente risponde con un orario non valido. (UC4.6)

3.5.6 UC4.6

Orario della riunione non valido.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'orario comunicato al chatbot non è valido.
- **Precondizione:** il dipendente ha comunicato al chatbot l'orario della riunione in un formato non valido o questo non è un orario lavorativo.
- **Post condizione:** il chatbot ha comunicato al dipendente che l'orario inserito non è valido.
- **Scenario principale:** il chatbot comunica all'utente che l'orario inserito non è valido tramite un messaggio testuale del tipo *"L'orario inserito non è valido"*.

3.5.7 UC4.7

Inserimento dei membri con i quali varrà effettuata la riunione.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente comunica al chatbot i membri con i quali verrà effettuata la riunione.
- **Precondizione:** il dipendente ha espresso la volontà di voler creare una nuova riunione al chatbot.
- **Post condizione:** il dipendente ha comunicato al chatbot i membri della riunione.
- **Scenario principale:** il chatbot chiede all'utente di elencare i membri che partecipano alla riunione con un messaggio del tipo *"Chi partecipa alla riunione?"*.
Il dipendente risponde comunicando al chatbot i membri che parteciperanno alla riunione.
- **Estensione:** il dipendente risponde con una lista di membri non valida. (UC4.8)

3.5.8 UC4.8

I membri che partecipano alla riunione sono non validi.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** la lista dei membri comunicata al chatbot non è valida.
- **Precondizione:** il dipendente ha comunicato al chatbot la lista dei membri che partecipano alla riunione in un formato non valido o essa contiene membri non validi.
- **Post condizione:** il chatbot ha comunicato al dipendente che la lista dei membri inserita non è valida.
- **Scenario principale:** il chatbot comunica all'utente che la lista dei membri inserita non è valida tramite un messaggio testuale del tipo *"La lista dei partecipanti inserita contiene membri non validi o è in un formato non valido"*.

3.6 UC5

Apertura del cancello aziendale per dipendenti

- **Attori:** utente autenticato tramite API key.
- **Descrizione:** un dipendente vuole aprire il cancello della sede aziendale desiderata attraverso l'applicativo di gestione del cancello, ossia il *broker MQTT_G*.
- **Precondizione:** un dipendente deve ancora aprire il cancello della sede aziendale attraverso l'applicativo.

- **Post condizione:** il dipendente è riuscito ad aprire il cancello aziendale attraverso l'applicativo.
- **Scenario principale:** se il dipendente non si è ancora autenticato nell'applicativo deve effettuare la procedura di *login_G* (UC1). A questo punto il dipendente invia un messaggio del tipo "*Apri il cancello della sede di XY*". Il sistema riceve il messaggio e richiede delle informazioni mancanti qualora non fossero presenti nel messaggio originale. Una volta inserite tali informazioni, il sistema manda il messaggio sul *broker MQTT_G* e conferma l'invio del comando. Le informazioni richieste sono le seguenti:
 - la sede della quale si vuole aprire il cancello. (UC5.1)
- **Estensioni:**
 - il *ChatBot_G* non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).
 - la sede inserita non esiste. (UC10).

3.6.1 UC5.1

Inserimento sede della quale si vuole aprire il cancello.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente comunica al chatbot la sede della quale vuole aprire il cancello.
- **Precondizione:** il dipendente ha espresso la volontà di voler aprire il cancello di una sede aziendale.
- **Post condizione:** il dipendente ha comunicato al chatbot la sede.
- **Scenario principale:** il chatbot chiede all'utente in quale sede vuole aprire il cancello con un messaggio del tipo "*Di quale sede vuoi aprire il cancello?*". Il dipendente risponde comunicando al chatbot la sede.
- **Estensione:** il dipendente risponde con una sede non valida. (UC5.2)

3.6.2 UC5.2

La sede inserita non è valida.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** la sede comunicata al chatbot non è valida.
- **Precondizione:** il dipendente ha comunicato al chatbot la sede in cui vuole aprire il cancello in un formato non valido.
- **Post condizione:** il chatbot ha comunicato al dipendente che la sede inserita non è valida.
- **Scenario principale:** il chatbot comunica all'utente che la sede inserita non è valida tramite un messaggio testuale del tipo "*La sede inserita non è valida*".

3.7 UC6

Servizio di ricerca documentale

- **Attori:** utente autenticato tramite API key.
- **Descrizione:** attraverso questa funzionalità un dipendente vuole ricercare dei documenti sul *repository_G* aziendale.
- **Precondizione:** un dipendente deve ancora effettuare l'operazione di ricerca sul *repository_G* aziendale.

- **Post condizione:** il sistema ha restituito un elenco di documenti trovati di interesse per il dipendente.
- **Scenario principale:** se il dipendente non si è ancora autenticato nell'applicativo deve effettuare la procedura di $login_G$ (UC1). A questo punto il dipendente invia un messaggio del tipo "*Cerca i documenti del progetto XY*". Il sistema riceve il messaggio e richiede delle informazioni mancanti qualora non fossero presenti nel messaggio originale. Una volta inserite tali informazioni, il sistema utilizza le API_G dell'applicativo per effettuare l'operazione di ricerca e alla fine restituisce di documenti con i rispettivi link. Le informazioni richieste sono le seguenti:
 - il nome del progetto nel quale effettuare la ricerca documentale. (UC6.1)
 - il nome del documento da ricercare. (UC6.3)
- **Estensioni:**
 - il $ChatBot_G$ non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).
 - il progetto inserito non esiste. (UC11)

3.7.1 UC6.1

Inserimento del nome del progetto di cui si vuole cercare i documenti.

- **Attori:** utente autenticato tramite API key_G .
- **Descrizione:** un dipendente comunica al chatbot il progetto di cui vuole cercare i documenti.
- **Precondizione:** il dipendente ha espresso la volontà di voler effettuare una ricerca documentale.
- **Post condizione:** il dipendente ha comunicato al chatbot il progetto.
- **Scenario principale:** il chatbot chiede all'utente in quale progetto vuole cercare i documenti con un messaggio del tipo "*In quale progetto vuoi cercare i documenti?*". Il dipendente risponde comunicando al chatbot il progetto.
- **Estensione:** il dipendente risponde con un nome di progetto non valido. (UC6.2)

3.7.2 UC6.2

Il nome del progetto inserito non è valido.

- **Attori:** utente autenticato tramite API key_G .
- **Descrizione:** il nome del progetto comunicato al chatbot non è valido.
- **Precondizione:** il dipendente ha comunicato al chatbot il nome del progetto in cui vuole effettuare la ricerca documentale in un formato non valido.
- **Post condizione:** il chatbot ha comunicato al dipendente che il progetto inserito non è valido.
- **Scenario principale:** il chatbot comunica all'utente che il progetto inserito non è valido tramite un messaggio testuale del tipo "*Il progetto inserito non è valido*".

3.7.3 UC6.3

Inserimento del nome del documento da ricercare.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente comunica al chatbot il nome del documento da ricercare.
- **Precondizione:** il dipendente ha espresso la volontà di voler effettuare una ricerca documentale.
- **Post condizione:** il dipendente ha comunicato al chatbot il nome del documento.
- **Scenario principale:** il chatbot chiede all'utente quale documento sta cercando con un messaggio del tipo "*Qual è il nome del documento?*".
Il dipendente risponde comunicando al chatbot il nome del documento.
- **Estensione:** il dipendente risponde con un nome di documento non valido. (UC6.4)

3.7.4 UC6.4

Il nome del documento inserito non è valido.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** il nome del documento comunicato al chatbot non è valido.
- **Precondizione:** il dipendente ha comunicato al chatbot il nome del documento che sta ricercando in un formato non valido o esso non è presente nel progetto selezionato.
- **Post condizione:** il chatbot ha comunicato al dipendente che il nome del documento inserito non è valido.
- **Scenario principale:** il chatbot comunica all'utente che il documento inserito non è valido tramite un messaggio testuale del tipo "*Il documento inserito non è valido*".

3.8 UC7

Servizio di creazione ticket_G

- **Attori:** utente autenticato tramite API key_G, applicativo esterno Redmine_G
- **Descrizione:** un dipendente vuole creare un ticket_G per tracciare bug_G legati ai progetti.
- **Precondizione:** il dipendente ha già effettuato il login_G(UC1) e si trova nella chat, deve ancora creare il ticket_G di suo interesse sull'applicativo Redmine_G.
- **Post condizione:** il ticket_G viene creato con successo e il ChatBot_G notifica l'utente del successo dell'operazione.
- **Scenario principale:** se il dipendente non si è ancora autenticato nell'applicativo deve effettuare la procedura di login_G (UC1). A questo punto il dipendente invia un messaggio del tipo "*Voglio creare un ticket*". Il sistema riceve il messaggio e richiede delle informazioni aggiuntive per completare l'operazione. Una volta inserite tali informazioni, il sistema, in base allo status code_G di risposta di Redmine_G,notifica l'utente dell'esito dell'operazione. Le informazioni richieste sono le seguenti:
 - l'oggetto del ticket_G, ovvero la motivazione di creazione dello stesso (UC7.1)
 - la descrizione del ticket_G (UC7.2)
 - lo status del ticket_G (UC7.3)
 - la priorità del ticket_G (UC7.4)
- **Estensioni:**
 - il ChatBot_G non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).

3.8.1 UC7.1

Inserimento dell'oggetto del *ticket_G*

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente ha segnalato la volontà di creare un nuovo *ticket_G* ed il *Chatbot_G* ha richiesto informazioni riguardo l'oggetto di questo.
- **Precondizione:** l'utente ha richiesto la creazione di un nuovo *ticket_G*
- **Post condizione:** l'utente ha comunicato al *Chatbot_G* l'oggetto da inserire nel nuovo *ticket_G*
- **Scenario principale:**
 - Il *ChatBot_G* chiede "Qual è l'oggetto del ticket che vuoi creare?".
 - l'utente inserisce tramite *chat_G* l'oggetto del *ticket_G*
- **Estensioni:** il dipendente inserisce un oggetto in un formato non valido. (UC7.5)

3.8.2 UC7.2

Inserimento di una descrizione del *ticket_G*

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente ha segnalato la volontà di creare un nuovo *ticket_G* ed il *Chatbot_G* ha richiesto informazioni riguardo la descrizione di questo.
- **Precondizione:** l'utente ha richiesto a creazione di un nuovo *ticket_G* ed ha già inserito l'oggetto di questo.
- **Post condizione:** l'utente ha comunicato al *Chatbot_G* la descrizione del nuovo *ticket_G*
- **Scenario principale:**
 - Il *ChatBot_G* chiede "Potresti fornirmi una descrizione del problema per cui vuoi creare il ticket?".
 - l'utente inserisce tramite *chat_G* la descrizione del *ticket_G*
- **Estensioni:** il dipendente inserisce una descrizione in un formato non valido. (UC7.6)

3.8.3 UC7.3

Inserimento dello status del *ticket_G*

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente ha segnalato la volontà di creare un nuovo *ticket_G* ed il *Chatbot_G* ha richiesto informazioni riguardo lo status di questo.
- **Precondizione:** l'utente ha richiesto a creazione di un nuovo *ticket_G* ed ha già inserito l'oggetto e descrizione di questo.
- **Post condizione:** l'utente ha comunicato al *Chatbot_G* lo status del nuovo *ticket_G*
- **Scenario principale:**
 - Il *ChatBot_G* chiede "Che status vuoi impostare per il ticket?".
 - l'utente inserisce tramite *chat_G* lo status del *ticket_G*
- **Estensioni:** il dipendente inserisce uno status in un formato non valido. (UC7.7)

3.8.4 UC7.4

Inserimento della priorità del *ticket_G*

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente ha segnalato la volontà di creare un nuovo *ticket_G* ed il *Chatbot_G* ha richiesto informazioni riguardo la priorità di questo.
- **Precondizione:** l'utente ha richiesto a creazione di un nuovo *ticket_G* ed ha già inserito l'oggetto, descrizione e status di questo.
- **Post condizione:** l'utente ha comunicato al *Chatbot_G* la priorità del nuovo *ticket_G*
- **Scenario principale:**
 - Il *ChatBot_G* chiede *"Che livello di priorità vuoi dare al ticket?"*.
 - l'utente inserisce tramite *chat_G* la priorità del *ticket_G*
- **Estensioni:** il dipendente inserisce una priorità in un formato non valido. (UC7.8)

3.8.5 UC7.5

Oggetto del *ticket_G* non valido

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente ha inserito l'oggetto del *ticket_G* ma i dati inseriti sono in un formato non valido.
- **Precondizione:** l'utente ha richiesto a creazione di un nuovo *ticket_G* ed ha inserito un oggetto in un formato non valido.
- **Post condizione:** Il *Chatbot_G*, dopo aver controllato i dati inseriti, notifica l'utente dell'errore e richiede di inserire nuovamente l'oggetto del *ticket_G*.
- **Scenario principale:**
 - Il *ChatBot_G* chiede *"L'oggetto inserito è in un formato non valido. Inserisci nuovamente un oggetto per il ticket."*.
 - l'utente inserisce tramite *chat_G* l'oggetto del *ticket_G*.

3.8.6 UC7.6

Descrizione del *ticket_G* non valida

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente ha inserito la descrizione inserente al nuovo *ticket_G* ma i dati inseriti sono in un formato non valido.
- **Precondizione:** l'utente ha richiesto a creazione di un nuovo *ticket_G* ed ha inserito una descrizione in un formato non valido.
- **Post condizione:** Il *Chatbot_G*, dopo aver controllato i dati inseriti, notifica l'utente dell'errore e richiede di inserire nuovamente la descrizione del *ticket_G*.
- **Scenario principale:**
 - Il *ChatBot_G* chiede *"La descrizione inserita è in un formato non valido. Inserisci nuovamente una descrizione per il ticket."*.
 - l'utente inserisce tramite *chat_G* la descrizione del *ticket_G*.

3.8.7 UC7.7

Status del $Ticket_G$ non valido

- **Attori:** utente autenticato tramite API key $_G$.
- **Descrizione:** l'utente ha inserito lo status inserente al nuovo $ticket_G$ ma i dati inseriti sono in un formato non valido.
- **Precondizione:** l'utente ha richiesto a creazione di un nuovo $ticket_G$ ed ha inserito uno status in un formato non valido.
- **Post condizione:** Il $Chatbot_G$, dopo aver controllato i dati inseriti, notifica l'utente dell'errore e richiede di inserire nuovamente lo status del $ticket_G$.
- **Scenario principale:**
 - Il $ChatBot_G$ chiede *"Lo status inserito è in un formato non valido. Inserisci nuovamente lo status per il ticket."*.
 - l'utente inserisce tramite $chat_G$ lo status del $ticket_G$.

3.8.8 UC7.8

Priorità del $ticket_G$ non valida

- **Attori:** utente autenticato tramite API key $_G$.
- **Descrizione:** l'utente ha inserito la priorità inserente al nuovo $ticket_G$ ma i dati inseriti sono in un formato non valido.
- **Precondizione:** l'utente ha richiesto a creazione di un nuovo $ticket_G$ ed ha inserito una priorità in un formato non valido.
- **Post condizione:** Il $Chatbot_G$, dopo aver controllato i dati inseriti, notifica l'utente dell'errore e richiede di inserire nuovamente la priorità del $ticket_G$.
- **Scenario principale:**
 - Il $ChatBot_G$ chiede *"La priorità inserita è in un formato non valido. Inserisci nuovamente la priorità per il ticket."*.
 - l'utente inserisce tramite $chat_G$ la priorità del $ticket_G$.

3.9 UC8

Logout $_G$

- **Attori:** utente autenticato tramite API key $_G$.
- **Descrizione:** l'utente autenticato deve poter effettuare l'operazione di $logout_G$ dal sistema.
- **Precondizione:** l'utente è autenticato ed il sistema mostra un pulsante in cui l'utente può eseguire l'operazione di $logout_G$.
- **Post condizione:** l'utente non è più autenticato all'interno del sistema.
- **Scenario principale:** l'utente digiterà *"Vorrei uscire"*: il chatbot effettuerà la domanda: *"Vuoi davvero effettuare il logout?"* e, come conferma, l'utente dovrà digitare *"Si"*, dove poi si troverà nello stato di $login_G$ nel sistema.
- **Estensioni:**
 - il $ChatBot_G$ non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).

3.10 UC9

Messaggio non riconosciuto dal *ChatBot_G*

- **Attori:** utente non autenticato e utente autenticato tramite API key_G.
- **Descrizione:** il *ChatBot_G* non riesce a comprendere un messaggio inviato dall'utente.
- **Precondizione:** l'utente manda un messaggio tramite *chat_G* al *ChatBot_G*.
- **Post condizione:** il *ChatBot_G* rileva un errore nel messaggio e notifica l'errore all'utente.
- **Scenario principale:**
 - l'utente manda un messaggio tramite *chat_G* al *ChatBot_G*.
 - il *ChatBot_G* cerca di interpretare il messaggio ma rileva un errore.
 - il *ChatBot_G* risponde all'utente segnalando l'errore nel messaggio.

3.11 UC10

Sede inesistente.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente ha inserito una sede che risulta inesistente.
- **Precondizione:** l'utente ha inserito una sede non esistente.
- **Post condizione:** il *ChatBot_G* notifica l'utente che la sede inserita non esiste.
- **Scenario principale:** Il *ChatBot_G* manda il messaggio "*La sede inserita non esiste.*".

3.12 UC11

Progetto inesistente

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente inserisce un codice per un progetto che risulta inesistente.
- **Precondizione:** l'utente ha inserito un codice per un progetto che risulta inesistente.
- **Post condizione:** il *ChatBot_G* notifica l'utente che il codice del progetto non esiste.
- **Scenario principale:** Il *ChatBot_G* manda il messaggio "*Il progetto con il codice XY non esiste.*".

3.13 UC12

Visualizzazione ore da consuntivare rimanenti

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente riceve il numero di ore da consuntivare rimanenti nella giornata.
- **Precondizione:** l'utente ha consuntivato un'attività giornaliera con successo.
- **Post condizione:** il *ChatBot_G* comunica all'utente il numero di ore da consuntivare rimanenti nella giornata.
- **Scenario principale:** il *ChatBot_G* comunica all'utente il numero di ore da consuntivare rimanenti nella giornata con il messaggio "*Ti mancano N ore da consuntivare oggi.*".

3.14 UC13

Verifica tracciamento presenza.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente vuole conoscere il suo stato attuale: se il tracciamento presenza è attivo o meno (e in che sede è attivo).
- **Precondizione:** il dipendente ha già effettuato il login_G(UC1) e si trova nella *chat*, deve ancora chiedere lo stato del tracciamento presenza.
- **Post condizione:** il dipendente è riuscito ad avere i dettagli sul suo stato di tracciamento presenza.
- **Scenario principale:** se il dipendente non si è ancora autenticato nell'applicativo deve effettuare la procedura di login_G (UC1). A questo punto il dipendente invia un messaggio del tipo "*Verifica stato tracciamento*". Il sistema riceve il messaggio e tramite richiesta API_G, recupera le informazioni sul tracciamento della presenza (UC2).
- **Estensioni:**
 - il ChatBot_G non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).

3.14.1 UC13.1

Tracciamento della presenza attivo.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente comunica al chatbot la necessità di recuperare le informazioni riguardo al suo tracciamento presenza. Questo restituisce l'informazione del tracciamento attivo.
- **Precondizione:** il dipendente ha espresso la volontà di recuperare le informazioni sul suo tracciamento presenza.
- **Post condizione:** il dipendente visualizza le informazioni riguardanti il suo tracciamento presenza.
- **Scenario principale:** il chatbot recupera tramite API_G le informazioni riguardanti il tracciamento presenza. Mostra poi al dipendente la sede e l'orario del check-in (UC2.1).

3.14.2 UC13.2

Tracciamento della presenza non attivo.

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** un dipendente comunica al chatbot la necessità di recuperare le informazioni riguardo al suo tracciamento presenza. Questo restituisce l'informazione che non è attivo nessun tracciamento.
- **Precondizione:** il dipendente ha espresso la volontà di recuperare le informazioni sul suo tracciamento presenza.
- **Post condizione:** il dipendente visualizza l'informazione che non ci sono tracciamenti attivi.
- **Scenario principale:** il chatbot recupera tramite API_G le informazioni riguardanti il tracciamento presenza. Non trovando nessun check-in attivo (l'ultima operazione risale ad un check-out UC2.2), restituisce un messaggio del tipo "*Nessun tracciamento presenza attivo*".

3.15 UC14

Annullamento operazione

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** l'utente vuole annullare un'operazione non ancora completata.
- **Precondizione:** l'utente sta effettuando un'operazione tramite *chat_G* e richiede al *ChatBot_G* l'annullamento di tale operazione.
- **Post condizione:** Il *ChatBot_G* annulla l'operazione che era stata richiesta.
- **Scenario principale:** se l'utente non si è ancora autenticato nell'applicativo deve effettuare la procedura di *login_G* (UC1). A questo punto l'utente invia un messaggio del tipo "*Annulla l'operazione*". Il *ChatBot_G* notifica quindi l'utente dell'avvenuto annullamento tramite il messaggio "*Operazione annullata*".

3.16 UC15

Login su *Piattaforma riunioni_G* esterna

- **Attori:** utente autenticato tramite API Key_G ma non ancora loggato sulla *Piattaforma riunioni_G*.
- **Descrizione:** l'utente vuole autenticarsi sulla *Piattaforma riunioni_G* esterna per inserirne una.
- **Precondizione:** l'utente non è autenticato nella *Piattaforma riunioni_G* esterna e vuole autenticarsi tramite inserimento dell'API Key_G univoca fornita da Imola Informatica.
- **Post condizione:** l'utente si è autenticato con successo all'interno della *Piattaforma riunioni_G* tramite l'inserimento dell'API Key_G e quindi può usufruire delle varie funzionalità all'interno.
- **Scenario principale:** l'utente inserisce l'API Key_G personale fornita da Imola Informatica per effettuare il *login_G* alla *Piattaforma riunioni_G*.
- **Estensioni:**
 - il *ChatBot_G* non riesce ad interpretare la richiesta (UC9).
 -
 - **Estensioni:**
 - l'operazione non va a buon fine (UC16).
 - l'utente inserisce una API Key_G in un formato non valido (UC15.1).
 - l'utente inserisce una API Key_G errata o non esistente (UC15.2).

3.16.1 UC15.1

API Key_G non valida.

- **Attori:** utente autenticato tramite API Key_G ma non ancora loggato sulla *Piattaforma riunioni_G*.
- **Descrizione:** l'API Key_G inserita dall'utente è espressa in un formato non valido.
- **Precondizione:** l'utente non è loggato sulla *Piattaforma riunioni_G* esterna e ha inserito un'API Key_G in un formato non valido.
- **Post condizione:** il *ChatBot_G*, dopo aver controllato i dati inseriti, notifica l'utente dell'errore e richiede di inserire nuovamente l'API Key_G corretta.
- **Scenario principale:**
 - Il *ChatBot_G* chiede "*L'API Key che hai inserito risulta espressa in un formato non valido. Per effettuare il login inserisci nuovamente l'API Key*".
 - L'utente inserisce nuovamente l'API Key_G tramite la *chat_G*.

3.16.2 UC15.2

API Key_G errata.

- **Attori:** utente autenticato tramite API Key_G ma non ancora loggato sulla *Piattaforma riunioni_G*.
- **Descrizione:** l'API Key_G inserita dall'utente è errata o non esistente.
- **Precondizione:** l'utente non è sulla *Piattaforma riunioni_G* esterna e ha inserito un'API Key_G errata o non esistente.
- **Post condizione:** il *ChatBot_G* notifica l'utente che l'API Key_G inserita è errata.
- **Scenario principale:** Il *ChatBot_G* manda il messaggio *"L'API Key che hai inserito risulta errata. Per effettuare il login inserisci API Key valida"*.

3.17 UC16

Visualizzazione errore impossibile completare operazione

- **Attori:** utente autenticato tramite API key_G.
- **Descrizione:** il *ChatBot_G* notifica l'utente che a causa un errore nel sistema non è possibile completare un'operazione.
- **Precondizione:** l'utente ha fatto una richiesta tramite *chat_G* che non può essere completata a causa di un errore del sistema.
- **Post condizione:** Il *ChatBot_G* notifica l'utente dell'errore.
- **Scenario principale:** Il *ChatBot_G* notifica l'utente dell'errore tramite il messaggio *"A causa di un errore ignoto non è possibile completare l'azione che hai richiesto."*.

3.18 UC17

Visualizzazione lista riunioni giornaliera

- **Attori:** utente autenticato tramite API key_G e che ha effettuato il login su una piattaforma riunioni.
- **Descrizione:** un dipendente vuole visualizzare la lista delle riunioni giornaliere.
- **Precondizione:** l'utente ha effettuato il login nel sistema tramite API key_G e nella piattaforma riunioni scelta dall'azienda.
- **Post condizione:** il *ChatBot_G* restituisce all'utente la lista delle riunioni giornaliere.
- **Scenario principale:** se il dipendente non si è ancora autenticato nell'applicativo deve effettuare la procedura di *login_G* (UC1). A questo punto il dipendente invia un messaggio del tipo *"Voglio vedere le riunioni di oggi"*. Il sistema utilizza le *API_G* dell'applicativo per effettuare l'operazione di ricerca e alla fine restituisce la lista di tutte le riunioni giornaliere.
- **Estensioni:**
 - il *ChatBot_G* non riesce ad interpretare la richiesta (UC9).
 - l'operazione non va a buon fine (UC16).

4 Requisiti

4.1 Requisiti funzionali

ID	Descrizione	Fonti
R1F1	un utente deve essere in grado di collegarsi all'applicativo	UC1, interna
R1F2	un utente deve essere in grado di eseguire un <i>login_G</i> tramite API key	UC1, capitolato
R1F3	un utente deve essere in grado di eseguire un <i>logout_G</i>	UC8, interna
R1F4	un utente deve essere in grado di inviare un messaggio testuale al bot	capitolato
R1F5	il bot deve essere in grado di recepire il messaggio ricevuto	interna
R1F6	il bot deve informare l'utente della mancata comprensione del messaggio	capitolato
R2F7	l'utente deve essere in grado di sapere della mancata comprensione del messaggio	interna
R1F8	il bot deve chiedere l'ora di inizio lavoro	UC3, capitolato
R2F9	il bot deve controllare il formato dell'ora inserita	interna
R2F10	l'utente deve essere in grado di capire se i dati inseriti sono validi	interna
R1F11	un utente deve essere in grado di inserire l'ora di inizio lavoro	UC3, capitolato
R1F12	il bot deve chiedere l'ora di fine lavoro	UC3, capitolato
R1F13	un utente deve essere in grado di inserire l'ora di fine lavoro	UC3, capitolato
R2F14	il bot deve controllare il formato dell'ora inserita	interna
R2F15	l'utente deve essere in grado di capire se i dati inseriti sono validi	interna
R1F16	il bot deve chiedere una descrizione del lavoro	UC3, interna
R1F17	un utente deve essere in grado di inserire una descrizione del lavoro	UC3, interna
R2F18	il bot deve chiedere la sala riunioni da prenotare	UC5, interna
R2F19	un utente deve essere in grado di inserire la sala riunione che intende prenotare	UC5, capitolato
R2F20	il bot deve controllare l'esistenza della sala riunioni	interna
R2F21	l'utente deve essere in grado di capire se i dati inseriti sono validi	UC5, capitolato

ID	Descrizione	Fonti
R2F22	il bot deve chiedere la data della riunione da prenotare	UC5, capitolato
R2F23	un utente deve essere in grado di inserire la data della riunione che intende prenotare	UC5, capitolato
R2F24	il bot deve controllare il formato della data inserita	interna
R2F25	l'utente deve essere in grado di capire se i dati inseriti sono validi	UC5, capitolato
R2F26	il bot deve chiedere l'ora della riunione da prenotare	UC5, interna
R2F27	un utente deve essere in grado di inserire l'ora della riunione che intende prenotare	UC5, capitolato
R2F28	il bot deve controllare il formato dell'ora inserita	interna
R2F29	l'utente deve essere in grado di capire se i dati inseriti sono validi	capitolato
R2F30	il bot deve informare l'utente se la prenotazione è andata a buon fine	capitolato
R2F31	il bot deve informare l'utente se la prenotazione non è andata a buon fine	UC5, capitolato
R2F32	l'utente deve essere in grado di capire l'esito della prenotazione	UC5, capitolato
R1F33	il bot deve chiedere all'utente il progetto da consuntivare	UC3, interna
R1F34	l'utente deve essere in grado di inserire il progetto da consuntivare	UC3, capitolato
R2F35	il bot deve controllare il progetto inserito	interna
R2F36	l'utente deve essere in grado di capire se i dati inseriti sono validi	UC3, capitolato
R1F37	il bot deve chiedere all'utente le ore da consuntivare	UC3, interna
R2F38	l'utente deve essere in grado di capire se i dati inseriti sono validi	UC3, capitolato
R1F39	l'utente deve essere in grado di inserire le ore da consuntivare	UC3, capitolato
R2F40	il bot deve controllare il formato dell'ora inserita	interna
R2F41	l'utente deve essere in grado di capire se i dati inseriti sono validi	UC3, capitolato
R1F42	il bot deve chiedere l'attività da consuntivare	UC3, interna
R1F43	l'utente deve essere in grado di inserire l'attività da consuntivare	UC3, capitolato
R2F44	il bot deve controllare l'attività inserita	interna
R2F45	l'utente deve essere in grado di capire se i dati inseriti sono validi	UC3, capitolato

ID	Descrizione	Fonti
R1F46	il bot deve chiedere il luogo di svolgimento dell'attività	UC3, interna
R1F47	l'utente deve essere in grado di comunicare il luogo di svolgimento dell'attività	UC3, capitolato
R2F48	il bot deve controllare il luogo inserito	interna
R2F49	l'utente deve essere in grado di capire se i dati inseriti sono validi	UC3, capitolato
R1F50	il bot deve informare l'utente delle ore giornaliere consuntivate	capitolato
R1F51	l'utente deve essere in grado di visualizzare le ore giornaliere consentite	capitolato
R1F52	il bot deve informare l'utente delle ore mancanti nella consuntivazione	capitolato
R1F53	l'utente deve essere in grado di visualizzare le ore mancanti nella consuntivazione	capitolato
R1F54	il bot deve registrare le operazioni nel sistema	capitolato
R2F55	l'utente deve essere in grado di eseguire un'operazione di undo	interna
R2F56	il bot deve chiedere su quale operazione eseguire un undo	interna
R2F57	il bot deve controllare l'operazione sulla quale eseguire un undo	interna
R2F58	l'utente deve essere in grado di capire se i dati inseriti sono validi	interna
R2F59	i dati scambiati tra applicativo e server devono essere criptati	interna
R1F60	l'utente deve poter registrare la sua presenza in una certa sede	UC2, capitolato
R1F61	l'utente deve poter effettuare il checkout	UC2, capitolato
R1F62	il chatbot deve richiedere conferma della sede da cui fare checkout	UC2, capitolato
R1F63	il chatbot deve richiedere all'utente in quale sede fare check-in	UC2, capitolato
R1F64	l'utente deve poter inserire la sede il cui fare il check-in	UC2, capitolato
R1F65	l'utente deve poter essere informato se inserisce una sede non valida	UC2
R3F66	l'utente deve essere in grado di chiedere al bot di aprire il cancello	UC4, capitolato
R3F67	il bot deve chiedere all'utente la sede del cancello da aprire	interna

ID	Descrizione	Fonti
R3F68	l'utente deve essere in grado di inserire la sede del cancello da aprire	UC4, capitolato
R3F69	il bot deve essere in grado di aprire il cancello	UC4, capitolato
R3F70	il bot deve informare l'utente di eventuali errori nell'apertura del cancello	UC4, interna
R3F71	l'utente deve essere in grado di visualizzare eventuali errori nell'apertura del cancello	UC4, interna
R3F72	il chatbot deve poter ricercare i documenti nell'archivio aziendale	UC6, capitolato
R3F73	l'utente deve poter comunicare al chatbot un ticket da creare	UC7, capitolato
R3F74	il chatbot deve chiedere all'utente l'oggetto del ticket da creare	UC7, capitolato
R3F75	l'utente deve poter inserire l'oggetto del ticket da creare	UC7, capitolato
R3F76	il chatbot deve richiedere una descrizione del ticket da creare	UC7, capitolato
R3F77	l'utente deve poter inserire una descrizione per il ticket da creare	UC7, capitolato
R3F78	il chatbot deve richiedere lo stato del ticket da creare	UC7, capitolato
R3F79	l'utente deve poter comunicare lo stato del ticket da creare	UC7, capitolato
R3F80	il chatbot deve chiedere all'utente la priorità del ticket da creare	UC7, capitolato
R3F81	l'utente deve poter inserire la priorità del ticket da creare	UC7, capitolato
R3F82	l'utente deve essere notificato se inserisce uno stato in un formato non valido	UC7
R3F83	l'utente deve essere notificato se inserisce un oggetto in un formato non valido	UC7
R3F84	il bot deve inserire il ticket nel sistema RedMine	UC7, capitolato
R3F85	l'utente deve poter comunicare al chatbot un documento da ricercare negli archivi aziendali	UC6, capitolato
R3F86	l'utente deve poter comunicare al chatbot che vuole visualizzare un elenco dei documenti di un progetto	UC6, capitolato
R3F87	il chatbot deve richiedere all'utente il nome del progetto di cui desidera visualizzare i documenti	UC6, capitolato
R3F88	l'utente deve poter inserire il nome del progetto di cui desidera i documenti	UC6, capitolato
R3F89	il chatbot deve richiedere all'utente il nome del documento cercato	UC6, capitolato

ID	Descrizione	Fonti
R3F90	l'utente deve poter comunicare il nome del documento che cerca	UC6, capitolato
R3F91	l'utente deve essere informato se inserisce un nome di un progetto non valido	UC6
R3F92	l'utente deve essere informato se inserisce un nome di un documento non valido	UC6

Tabella 1: Requisiti funzionali

4.2 Requisiti prestazionali

ID	Descrizione	Fonti
R1P1	L'architettura server dovrà disporre al massimo di 2 CPU e 1 Gi per istanza.	capitolato
R1P2	Le comunicazioni tra applicazione e server dovranno avvenire nel momento in cui l'utente invierà il messaggio.	capitolato
R2P3	Il server dovrà rispondere all'utente in tempi considerati ragionevoli per una chat, nel migliore dei casi real-time.	interna
R1P4	Il bot dovrà essere prevalentemente disponibile in orario lavorativo.	interna
R2P5	L'interfaccia dovrà essere più semplice possibile, per permettere all'utente di interagire con il minor numero di click.	interna

Tabella 2: Requisiti prestazionali

4.3 Requisiti qualitativi

ID	Descrizione	Fonti
R1Q1	i test eseguiti devono ricoprire almeno 80% del codice prodotto	capitolato
R1Q2	tutto il codice prodotto deve rispettare il documento Norme di Progetto _D	interna
R1Q3	tutto il codice prodotto deve rispettare il documento Piano di Qualifica _D	interna
R2Q4	il codice prodotto deve essere conservato su un <i>repository</i> _G remoto (es. GitHub _G)	interna
R1Q5	deve essere presente una documentazione su tutte le scelte adottate	interna

ID	Descrizione	Fonti
R2Q6	deve essere presente una documentazione sull'utilizzo dell'applicativo	capitolato

Tabella 3: Requisiti qualitativi

4.4 Requisiti di vincolo

ID	Descrizione	Fonti
R2V1	<p>l'applicativo deve poter essere eseguito su dispositivi desktop_G compatibili con browser di ultima generazione, con le seguenti caratteristiche minime:</p> <p>Sistema operativo</p> <ul style="list-style-type: none">• Windows 8• macOS 10.12• Ubuntu a 64 bit 18.04 <p>Caratteristiche Hardware</p> <ul style="list-style-type: none">• 512MB / 2GB di RAM per la versione a 64-bit• 200MB di spazio disco• Un processore Intel Pentium 4 o versioni successive che supporti SSE3 / Mac con Intel x86 o con processore Apple <p>Ref. https://support.google.com/chrome/a/answer/7100626?hl=it https://www.mozilla.org/en-US/firefox/101.0.1/system-requirements/</p>	interna
R2V2	<p>l'applicativo deve poter essere eseguito su dispositivi mobile_G con le seguenti caratteristiche minime:</p> <p>Sistema operativo</p> <ul style="list-style-type: none">• iOS 13• Android 6.0 <p>Ref. https://support.mozilla.org/en-US/kb/will-firefox-work-my-mobile-device</p>	interna
R1V3	il front-end dovrà essere sviluppato usando il framework _G React _G	interna
R2V4	il front-end dovrà seguire lo standard HTML5 _G	interna
R1V5	il back-end dovrà essere sviluppato usando il framework _G Django _G	interna

ID	Descrizione	Fonti
R2V6	l'applicativo deve essere accessibile anche per gli utenti con difficoltà visive ref. Web Content Accessibility Guidelines (WCAG) 2.1 https://www.w3.org/Translations/WCAG21-it/	interna
R2V7	l'applicativo dovrà essere supportato dai browser _G moderni: <ul style="list-style-type: none">• Google Chrome a partire da v100• Firefox a partire da v70• Safari a partire da v15 Questi permettono di adottare gli ultimi standard per i linguaggi HTML, CSS e JS.	interna
R1V8	in caso di qualsiasi errore l'applicativo non dovrà andare in crash	capitolato

Tabella 4: Requisiti di vincolo

5 Tracciamento requisiti

5.1 Requisito - Fonte

Requisito	Fonte
R1F1	UC1, interna
R1F2	UC1, capitolato
R1F3	UC8, interna
R1F4	capitolato
R1F5	interna
R1F6	capitolato
R2F7	interna
R1F8	UC3, capitolato
R2F9	interna
R2F10	interna
R1F11	UC3, capitolato
R1F12	UC3, capitolato
R1F13	UC3, capitolato
R2F14	interna
R2F15	interna
R1F16	UC3, interna
R1F17	UC3, interna
R2F18	UC5, interna
R2F19	UC5, capitolato
R2F20	interna
R2F21	UC5, capitolato
R2F22	UC5, capitolato
R2F23	UC5, capitolato
R2F24	interna
R2F25	UC5, capitolato
R2F26	UC5, interna
R2F27	UC5, capitolato
R2F28	interna
R2F29	capitolato
R2F30	capitolato
R2F31	UC5, capitolato
R2F32	UC5, capitolato

Requisito	Fonte
R1F33	UC3, interna
R1F34	UC3, capitolato
R2F35	interna
R2F36	UC3, capitolato
R1F37	UC3, interna
R2F38	UC3, capitolato
R1F39	UC3, capitolato
R2F40	interna
R2F41	UC3, capitolato
R1F42	UC3, interna
R1F43	UC3, capitolato
R2F44	interna
R2F45	UC3, capitolato
R1F46	UC3, interna
R1F47	UC3, capitolato
R2F48	interna
R2F49	UC3, capitolato
R1F50	capitolato
R1F51	capitolato
R1F52	capitolato
R1F53	capitolato
R1F54	capitolato
R2F55	interna
R2F56	interna
R2F57	interna
R2F58	interna
R2F59	interna
R1F60	UC2, capitolato
R1F61	UC2, capitolato
R1F62	UC2, capitolato
R1F63	UC2, capitolato
R1F64	UC2, capitolato
R1F65	UC2
R3F66	UC4, capitolato
R3F67	interna

Requisito	Fonte
R3F68	UC4, capitolato
R3F69	UC4, capitolato
R3F70	UC4, interna
R3F71	UC4, interna
R3F72	UC6, capitolato
R3F73	UC7, capitolato
R3F74	UC7, capitolato
R3F75	UC7, capitolato
R3F76	UC7, capitolato
R3F77	UC7, capitolato
R3F78	UC7, capitolato
R3F79	UC7, capitolato
R3F80	UC7, capitolato
R3F81	UC7, capitolato
R3F82	UC7
R3F83	UC7
R3F84	UC7, capitolato
R3F85	UC6, capitolato
R3F86	UC6, capitolato
R3F87	UC6, capitolato
R3F88	UC6, capitolato
R3F89	UC6, capitolato
R3F90	UC6, capitolato
R3F91	UC6
R3F92	UC6
R1P1	capitolato
R1P2	capitolato
R2P3	interna
R1P4	interna
R2P5	interna
R1Q1	capitolato
R1Q2	interna
R1Q3	interna
R2Q4	interna
R1Q5	interna



Requisito	Fonte
R2Q6	capitolato
R1V1	interna
R2V2	interna
R1V3	interna
R2V4	interna
R1V5	interna
R2V6	interna
R2V7	interna
R1V8	interna

Tabella 5: Requisito - Fonte

5.2 Fonte - Requisito

Fonte	Requisito
UC1	R1F1, R1F2
UC2	R1F60, R1F61, R1F62, R1F63, R1F64, R1F65
UC3	R1F8, R1F11, R1F12, R1F13, R1F16, R1F17, R1F33, R1F34, R1F37, R1F39, R1F42, R1F43, R1F46, R1F47, R2F36, R2F38, R2F41, R2F45, R2F49
UC4	R3F66, R3F68, R3F69, R3F70, R3F71
UC5	R2F18, R2F19, R2F21, R2F22, R2F33, R2F25, R2F26, R2F27, R2F31, R2F32
UC6	R3F72, R3F85, R3F86, R3F87, R3F88, R3F89, R3F90, R3F91, R3F92
UC7	R3F73, R3F74, R3F75, R3F76, R3F77, R3F78, R3F79, R3F80, R3F81, R3F82, R3F83, R3F84
UC8	R1F3
interna	R1F1, R1F3, R1F5, R1F16, R1F17, R1F33, R1F37, R1F42, R1F46, R1P4, R1Q2, R1Q3, R1Q5, R1V1, R1V3, R1V5, R1V8, R2F7, R2F9, R2F10, R2F14, R2F15, R2F18, R2F20, R2F24, R2F26, R2F28, R2F35, R2F40, R2F44, R2F48, R2F55, R2F56, R2F57, R2F58, R2F59, R2P3, R2P5, R2Q4, R2V2, R2V4, R2V6, R2V7, R3F67, R3F70, R3F71
capitolato	R1F2, R1F4, R1F6, R1F8, R1F11, R1F12, R1F13, R1F34, R1F39, R1F43, R1F47, R1F50, R1F51, R1F52, R1F53, R1F54, R1F60, R1F61, R1F62, R1F63, R1F64, R1P1, R1P2, R1Q1, R2F19, R2F21, R2F22, R2F23, R2F25, R2F27, R2F29, R2F30, R2F31, R2F32, R2F36, R2F38, R2F41, R2F45, R2F49, R2Q6, R3F66, R3F68, R3F69, R3F72, R3F73, R3F74, R3F75, R3F76, R3F77, R3F78, R3F79, R3F80, R3F81, R3F84, R3F85, R3F86, R3F87, R3F88, R3F89, R3F90

Tabella 6: Fonte - Requisito