



**UNIVERSITÀ
DI TRENTO**

**Dipartimento di
Ingegneria e Scienza dell'Informazione**

Progetto: TRENTUS

Analisi e Progettazione

INDICE

| | |
|---|-----------|
| Scopo del documento..... | 3 |
| 1. Requisiti funzionali..... | 4 |
| 2. Analisi dei componenti..... | 15 |
| 2.1. Definizione dei componenti..... | 15 |
| CMP1: Gestione autenticazione:..... | 15 |
| CMP2: Gestione database..... | 16 |
| CMP3: Gestione richieste di registrazione servizio..... | 16 |
| CMP4: Gestione registrazione GdS..... | 17 |
| CMP5: Gestione servizi..... | 17 |
| CMP6: Gestione avvisi..... | 18 |
| CMP7: Gestione delle segnalazioni..... | 19 |
| CMP8: Gestione grafici..... | 19 |
| CMP9: Gestione classifica..... | 20 |
| CMP10: Gestione form di segnalazione:..... | 20 |
| CMP11: Gestione sondaggi:..... | 21 |
| CMP12: Gestione feedback:..... | 22 |
| CMP13: Gestione registrazione utente..... | 22 |
| CMP14: Gestione utente..... | 23 |
| CMP15: Gestione GdS..... | 23 |
| CMP16: Filtro dati utente..... | 24 |
| 2.2 Diagramma dei componenti..... | 25 |
| 3. Diagramma delle classi..... | 26 |
| 1. Utente..... | 26 |
| 2. GdS..... | 26 |
| 3. SuperAdmin..... | 27 |
| 4. Servizio..... | 27 |
| 5. Schemi dei documenti..... | 27 |
| 6. Form..... | 28 |
| 7. Segnalazione..... | 28 |
| 8. Sondaggio..... | 29 |
| 9. Feedback..... | 29 |
| 10. Avvisi..... | 29 |
| 3.1 Diagramma delle classi..... | 30 |
| 4. Dal class diagram alle API..... | 31 |

Scopo del documento

Il presente documento riporta la specifica dei requisiti di sistema del progetto Trentus usando quanto più possibile diagrammi in Unified Modeling Language (UML). Nel precedente documento sono stati definiti gli obiettivi del progetto e i requisiti usando un linguaggio naturale. Ora i requisiti verranno specificati usando sia il linguaggio naturale che linguaggi più formali e strutturati, in particolare UML per la descrizione dei requisiti funzionali. Inoltre, tenendo conto di tali requisiti, viene presentato il design del sistema con l'utilizzo del diagramma dei componenti e un diagramma delle classi.

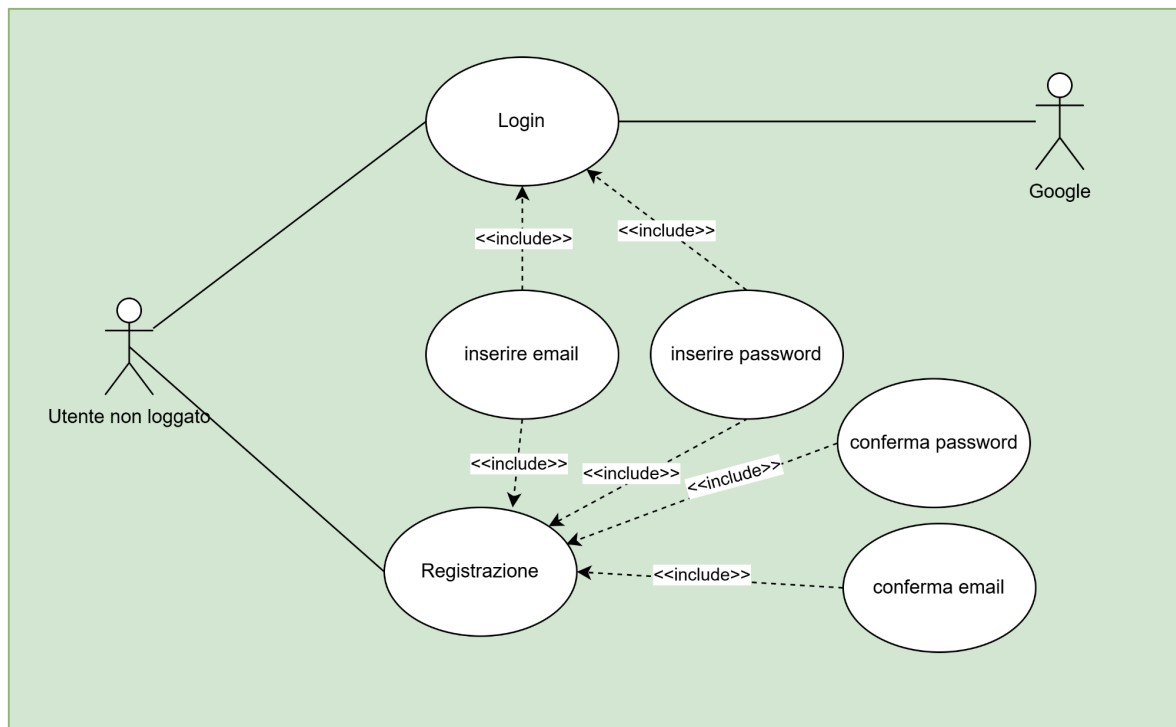
1. Requisiti funzionali

Nel presente capitolo vengono riportati i requisiti funzionali (RF) del sistema utilizzando il linguaggio naturale e Use case Diagram (UCD) scritti in UML.

Utente non loggato

RF1: Login

RF2: Registrazione Utente



Use Case RF1: Login

Riassunto:

Questo use case descrive come un utente non loggato effettua il login sul sito.

Descrizione:

1. L'utente visualizza la schermata di login.
2. Se l'utente inserisce all'interno degli appositi campi l'email e la password e clicca sul pulsante "Login" presente nella schermata, nel caso in cui l'email e la password sono corretti [eccezione 1] e l'account con tali dati è già stato registrato [eccezione 2], allora si aprirà la home page dell'applicazione.

Eccezioni:

1. Se l'utente inserisce una email sbagliata o non associata alla password inserita, l'applicazione restituisce un messaggio di errore e chiede nuovamente di effettuare il login.
2. Se l'utente inserisce le credenziali di un account non ancora registrato, l'applicazione restituisce un messaggio di errore e chiede nuovamente di effettuare l'accesso.

Use Case RF2: Registrazione

Riassunto:

Questo use case descrive come un utente che non dispone di un account può registrarsi a Trentus.

Descrizione:

1. L'utente visualizza la schermata di login.
2. L'utente clicca la voce "Non hai un account? Registrati" per accedere alla pagina di registrazione di un nuovo account.
3. L'utente inserisce negli appositi campi una email e una password ripetuta due volte. Se la password corrisponde in entrambi i campi [eccezione 1], allora il sistema procederà a mandare una email all'utente per verificare definitivamente l'account.

Eccezioni:

1. Se l'utente inserisce due password diverse nei due campi in cui è richiesta, la schermata mostrerà un messaggio di errore e chiederà all'utente di inserire un'unica password in entrambi i campi.

Utente

RF3: Visualizzazione di una lista di servizi, di cui è possibile verificarne lo stato.

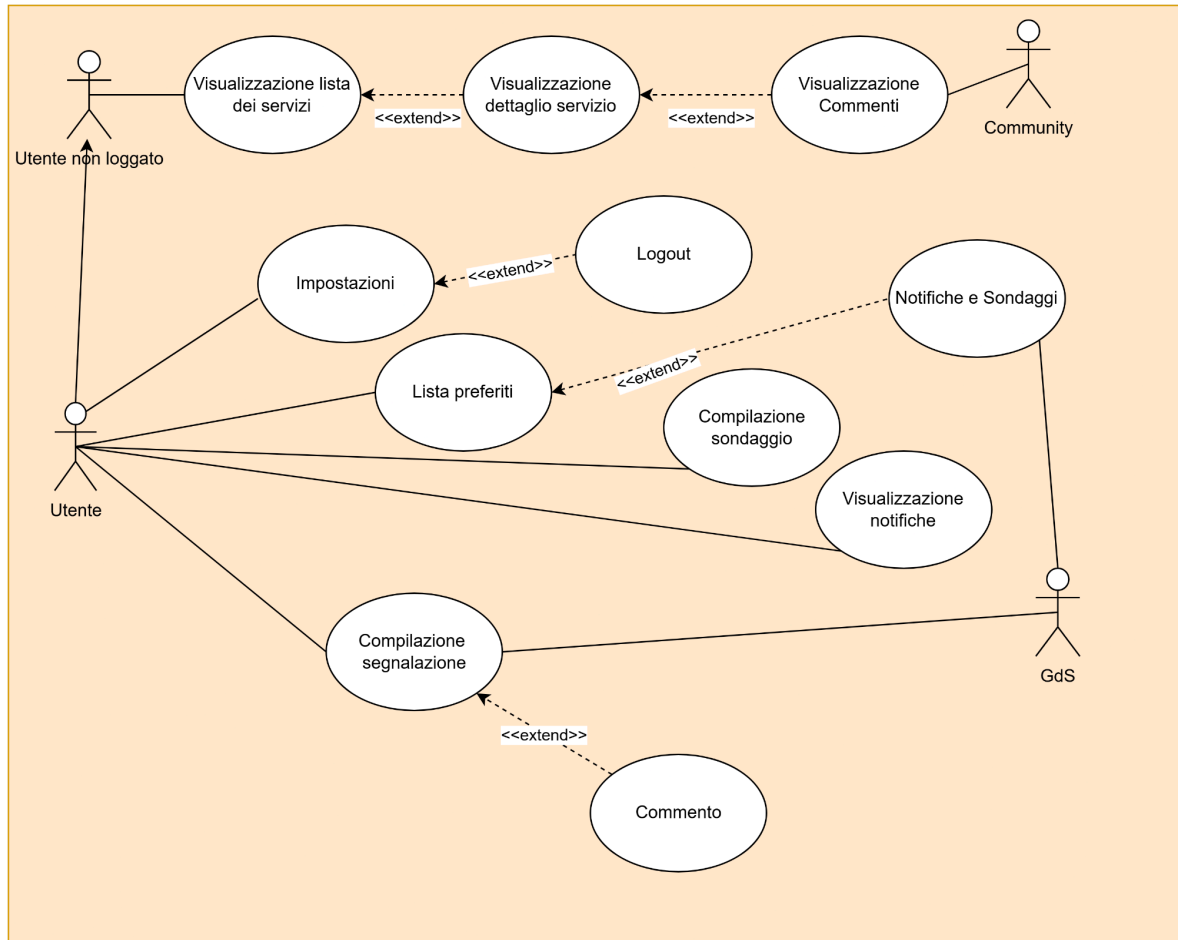
RF4: Possibilità di segnalare il malfunzionamento di un servizio e, eventualmente, di lasciare un commento alla segnalazione.

RF5: Visualizzazione di una lista dei soli servizi preferiti dell'utente, con la possibilità di ricevere delle notifiche push da questi servizi.

RF6: Impostazioni dell'applicazione.

RF7: Visualizzazione dettagli notifiche (avvisi).

RF8: Compilazione sondaggi.



Use case RF3: Visualizzazione lista dei servizi

Riassunto: Questo Use Case descrive la principale funzione della home page: la visualizzazione di una lista di servizi, di cui è possibile vederne lo stato attuale.

Descrizione:

1. L'utente entra sul sito di Trentus e viene reindirizzato alla Home Page.
2. L'utente può scrollare la pagina, guardando lo stato dei vari servizi presenti nella lista di Trentus.
3. L'utente può usare la barra di ricerca per cercare un servizio preciso della lista di Trentus. (NOTA: non implementata sul sito finale *vedi requisiti funzionali D1*)
4. L'utente può cliccare su uno specifico servizio della lista di Trentus per avere più dettagli [estensione 1] e guardare i commenti degli altri utenti [estensione 2].

Estensioni:

1. Se l'utente seleziona un servizio, saranno visibili delle informazioni più precise sullo stato del servizio.
2. Se l'utente seleziona un servizio, questo potrà visualizzare i commenti lasciati dagli altri utenti di Trentus riguardo a quello stesso servizio.

User Case RF4: Segnalazione del malfunzionamento di un servizio

Riassunto: Questo Use Case descrive il procedimento di segnalazione di un malfunzionamento riscontrato dall'utente.

Descrizione:

1. L'Utente visualizza la pagina dei dettagli di un servizio.
2. L'Utente loggato [eccezione 1] può usare il tasto "Segnala" per compilare la segnalazione di un problema del servizio.
3. L'Utente compila il questionario fornito dai Gestori del Servizio [estensione 1].

Eccezioni:

1. Se un utente non loggato clicca il tasto segnala, verrà reindirizzato alla pagina di Login/Registrazione.

Estensioni:

1. Se l'utente vuole, può lasciare un commento per specificare al meglio possibile il problema riscontrato.

User Case RF5: Lista Preferiti

Riassunto: Questo Use Case descrive la sezione Preferiti, accessibile dalla Home Page.

Descrizione:

1. L'Utente loggato accede alla schermata dei servizi preferiti cliccando sull'icona della stella presente nella Home Page.
2. Visualizzazione della lista di tutti i servizi segnati come "Preferiti" dall'utente, di cui è possibile vederne lo stato. [estensione 1].

Estensioni:

1. L'Utente riceve delle notifiche e dei sondaggi compilabili, visibili sulla sidebar di sinistra, mandate dai gestori dei servizi preferiti dell'utente.

User Case RF6: Impostazioni

Riassunto: L'Utente può cliccare sull'icona delle impostazioni per cambiare alcuni aspetti dell'applicazione.

Descrizione:

1. L'Utente, in qualsiasi schermata dell'applicazione, può cliccare sul tasto "Impostazioni".
2. Se l'Utente clicca sul tasto impostazioni, un menù a tendina comparirà al di sotto dell'icona stessa [estensione 1].

Estensioni

1. L'Utente, dopo aver evocato la tendina delle impostazioni, può effettuare il login (se l'Utente non è loggato) o il logout.

Use Case RF7: Visualizzazione dettagli notifiche

Riassunto: Questo Use Case descrive il sistema di notifiche per l'Utente.

Descrizione:

1. L'Utente, in qualsiasi schermata dell'applicazione, può cliccare su una notifica presente sulla barra laterale di sinistra [eccezione 1] [eccezione 2].
2. Se l'Utente clicca sulla notifica, apparirà una schermata contenente i dettagli di tale notifica.

Eccezioni:

1. Nel caso in cui l'Utente non abbia servizi preferiti, sulla barra laterale di sinistra non verranno riportate notifiche, in quanto gli utenti ricevono le notifiche solo dei propri servizi preferiti.
2. Nel caso in cui i servizi preferiti di un Utente non abbiano mandato alcun avviso, non ci sarà nessuna notifica su cui cliccare.

Use Case RF8: Compilazione sondaggi

Riassunto: Questo Use Case descrive il processo di compilazione dei sondaggi per l'Utente.

Descrizione:

1. L'Utente, in qualsiasi schermata dell'applicazione, può cliccare su un sondaggio presente sulla barra laterale di sinistra [eccezione 1] [eccezione 2].
2. Una volta compilato il sondaggio, l'utente cliccherà sul pulsante "Conferma". [eccezione 3].

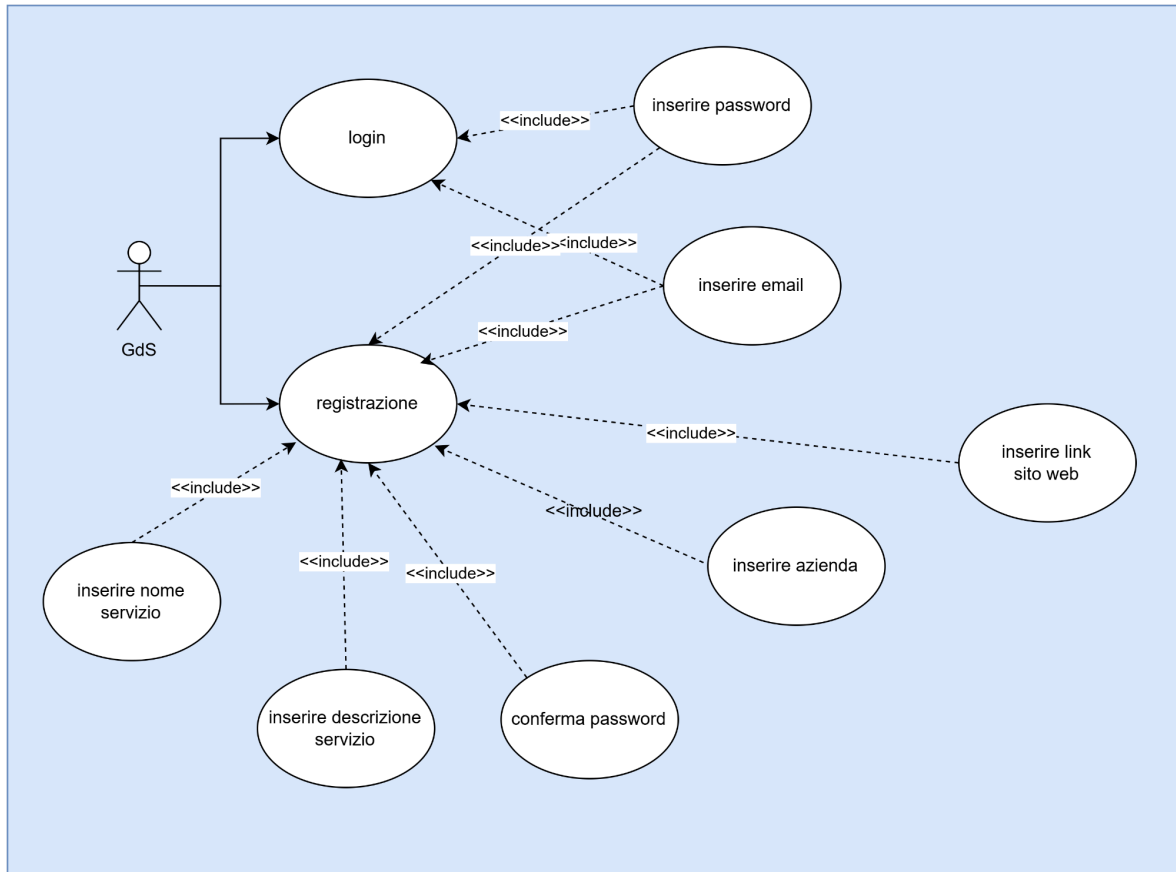
Eccezioni:

1. Nel caso in cui l'Utente non abbia servizi preferiti, sulla barra laterale di sinistra non verrà riportato alcun sondaggio, in quanto gli utenti ricevono sondaggi solo dei propri servizi preferiti.
2. Nel caso in cui i servizi preferiti di un Utente non abbiano offerto alcun sondaggio, non ci sarà nessuna sezione su cui cliccare.
3. Se l'Utente non compila il sondaggio ma clicca sul tasto conferma, verrà evocato un messaggio di errore.

Gestore del servizio - accesso alla piattaforma

RF9: Registrazione nuovo servizio

RF10: Login gestore del servizio



Use Case RF9: Registrazione nuovo servizio

Riassunto: Il gestore di un servizio può fare richiesta a Trentus di aggiungere il proprio servizio all'applicazione, registrandolo.

Descrizione:

1. L'Utente accede alla schermata di Login/Registrazione
2. L'Utente clicca sul link "Registra il tuo servizio su Trentus"
3. L'Utente viene reindirizzato alla pagina di Registrazione servizio
4. L'Utente inserisce i dati richiesti dalla pagina e conferma la registrazione [eccezione 1]
5. L'Utente verrà contattato da un Admin di Trentus, che verificherà le veridicità del servizio da registrare [eccezione 2]

Eccezioni:

1. Se i dati inseriti dall'Utente non sono corretti, non sarà possibile proseguire la registrazione del servizio.
2. Se l'Utente non riesce a fornire le informazioni richieste dall'Admin di Trentus non sarà possibile proseguire la registrazione del servizio.

Use Case RF10: Login del GdS - uguale a RF1

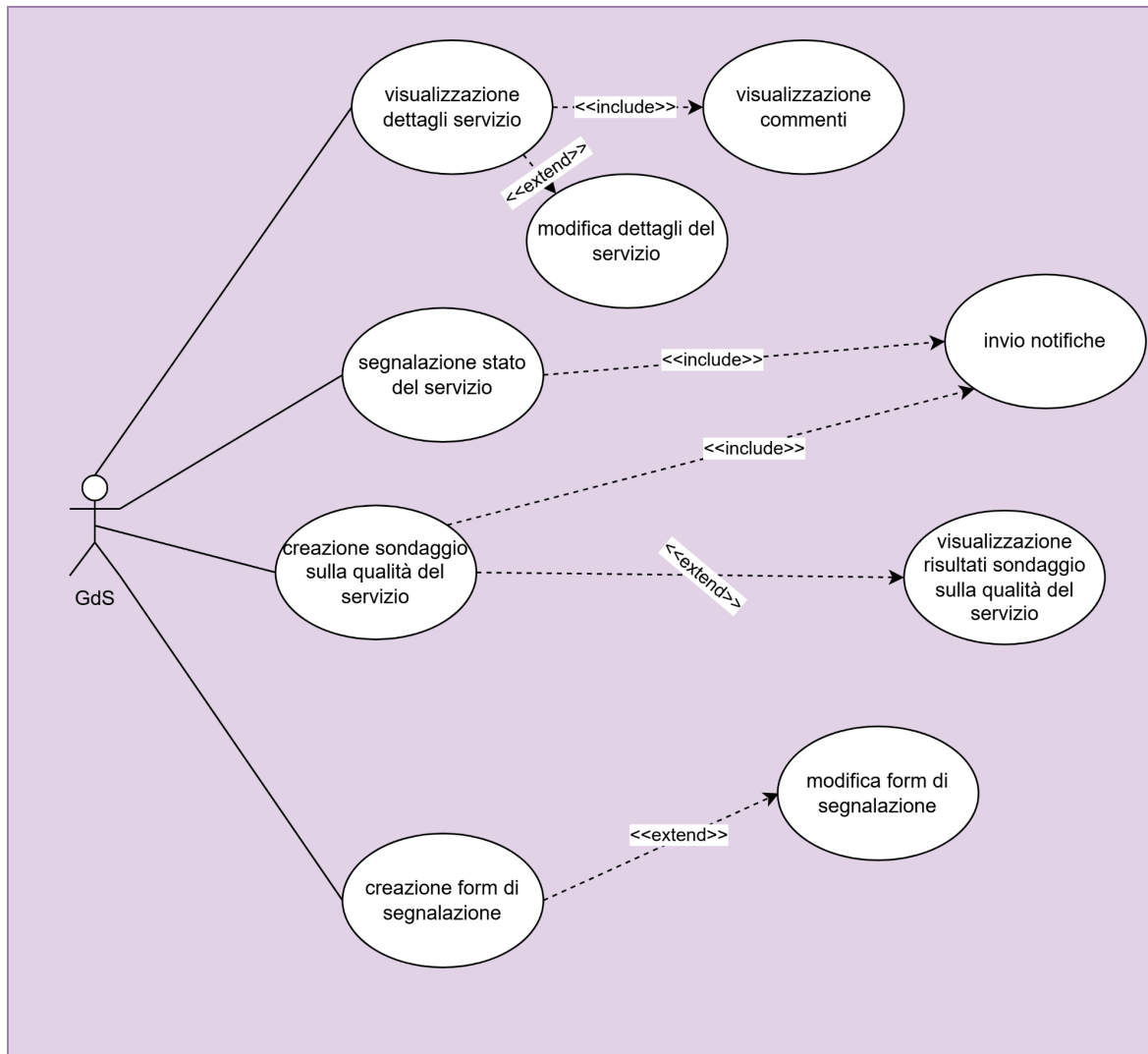
Gestore del servizio - uso della piattaforma

RF11: Visualizzazione dei dettagli del servizio

RF12: Segnalazione dello stato di un servizio

RF13: Creazione e modifica del sondaggio di segnalazione per gli utenti

RF14: Creazione e visualizzazione di sondaggio sulla qualità del servizio per gli utenti



Use Case RF11: Visualizzazione dettagli servizio

Riassunto: Questo Use Case descrive l'uso della dashboard principale messa a disposizione ai Gestori dei Servizi.

Descrizione:

1. L'Utente accede alla schermata principale di Trentus per i GdS [estensione 1]
2. L'Utente può visionare i dati sul proprio servizio, come numero di segnalazioni e percentuali sul motivo delle stesse [estensione 2]

Estensioni

1. L'Utente può cliccare sul tasto "Modifica" per modificare i dati del proprio servizio (nome, logo ecc...)
2. L'Utente può accedere alla sezione commenti per visionare i commenti lasciati dalla community

Use Case RF12: Segnalazione dello stato di un servizio

Riassunto: Questo Use Case descrive come il Gestore di un Servizio può notificare gli utenti di un malfunzionamento.

Descrizione:

1. L'Utente accede alla schermata principale di Trentus per i GdS
2. L'Utente clicca sul tasto "Invia Segnalazioni" presente nella navbar superiore
3. L'Utente può compilare il modulo di segnalazione, specificando tipo di servizio, tempi di attesa e dando una descrizione del problema. [eccezione 1]
4. L'Utente conferma l'invio della segnalazione [estensione 1]

Eccezioni:

1. Se il Gestore del Servizio inserisce dei dati sbagliati (oppure le scadenze non vengono rispettate), gli Utenti non avranno informazioni sulla cosa a meno che non venga fatta un' altra segnalazione dai GdS.

Estensioni:

1. Se il Gestore del Servizio invia una segnalazione, tutti gli Utenti che hanno inserito quel servizio tra i preferiti riceveranno una notifica sull'app.

Use Case RF13: Sondaggi sulla qualità del servizio degli utenti

Riassunto: Questo Use Case descrive come il Gestore di un Servizio può creare un sondaggio da far compilare ai propri utenti per reperire informazioni e consigli sulla qualità del servizio offerto.

Descrizione:

1. L'Utente accede alla schermata principale di Trentus per i GdS
2. L'Utente clicca sul tasto "Invia un sondaggio" visibile nella schermata Home
3. L'Utente accede alla schermata di creazione sondaggi sulla qualità del servizio
4. L'Utente, attraverso il sito, può creare dei sondaggi con varie modalità di input, tra cui Scelta Multipla, Checklist e Risposta Aperta.
5. L'Utente conferma la creazione del sondaggio [estensione 1] [eccezione 1]

Eccezioni:

1. Se il GdS conferma l'invio di un sondaggio vuoto, il sistema mostrerà un messaggio di errore.

Estensioni:

1. Non appena il GdS conferma la creazione del sondaggio sulla qualità del servizio, gli utenti che hanno tale servizio come "Preferito" riceveranno una notifica sull'app.

Use Case RF14: Creazione di un sondaggio di segnalazione

Riassunto: Questo Use Case descrive come viene creato il form di segnalazione che viene mostrato agli utenti quando decidono di segnalare una problematica del servizio.

Descrizione:

1. L'Utente clicca sul tasto "Aggiungi opzione al form" presente nell'Homepage.
2. L'Utente, grazie al tool di creazione sondaggi presente sull'app, crea il form.

Estensioni:

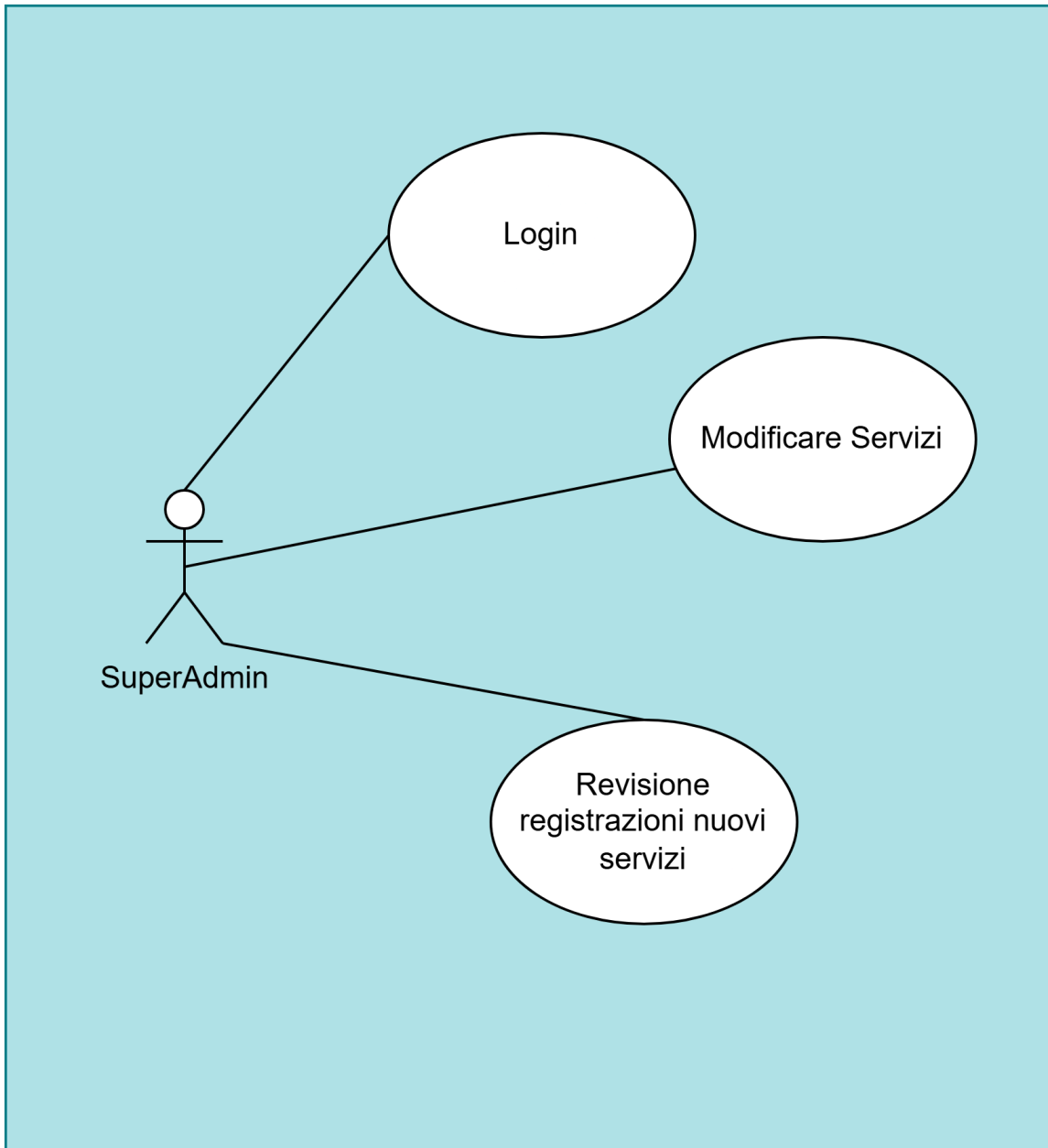
1. L'Utente può modificare il sondaggio di segnalazione precedentemente creato, eliminando delle opzioni precedentemente inserite o aggiungendole delle nuove.

SuperAdmin: Azioni della piattaforma

RF15: Login del SuperAdmin

RF16: Revisione delle richieste di aggiunta servizio

RF17: Modifica dei servizi



Use Case RF15: Login del SuperAdmin: Uguale a RF1

Use Case RF16: Revisione delle richieste di aggiunta servizio:

Riassunto: Questo Use Case descrive come il SuperAdmin gestisce le richieste di aggiunta di nuovo servizio da parte dei Gestori di Servizio.

Descrizione:

1. L'Utente accede alla schermata principale di Trentus per i SuperAdmin
2. L'Utente, attraverso la sidebar laterale delle notifiche, vede le richieste di aggiunta servizio in sospeso

3. L'Utente, cliccando sulla notifica, accede alla pagina di revisione della richiesta, dove vengono riportati i dati inseriti dal GdS durante il processo di registrazione del servizio
4. L'Utente, dopo aver confermato la veridicità dei dati forniti, può accettare o rifiutare la richiesta del GdS.

Use Case RF17: Modifica dei Servizi

Riassunto: Questo Use Case mostra come un SuperAdmin possa modificare i dettagli di un servizio presente sulla piattaforma

Descrizione:

1. L'Utente accede alla schermata principale di Trentus per i SuperAdmin
2. L'Utente può cliccare sull'icona dei tre puntini sulla scheda di un servizio e selezionare la voce "Modifica" per modificare i dettagli del servizio selezionato
3. L'Utente accede alla schermata di modifica dei dati del servizio, dove l'Utente può modificare i dati forniti dal GdS

2. Analisi dei componenti

Nel presente capitolo viene presentata l'architettura in termini di componenti (CMP) interni al sistema definiti sulla base dei requisiti analizzati in precedenza. Viene poi adottato l'uso del diagramma dei componenti per rappresentare l'interconnessione tra i vari componenti, identificando quindi le interfacce tra questi e verso sistemi esterni.

2.1. Definizione dei componenti

In questa sezione vengono descritti i componenti con le apposite interfacce previste (offerte e richieste) per Trentus.

CMP1: Gestione autenticazione:

Descrizione:

Il componente si occupa di gestire l'autenticazione dei differenti tipi di account del nostro sistema, ovvero: utente, GdS, Super Admin. Include le pagine per effettuare il login per ognuno dei ruoli previsti e relativi componenti grafici per il logout.

Il componente permette di identificare coloro che effettuano l'accesso alla web app, permettendogli di accedere alle funzionalità a loro riservate in base ai differenti ruoli.

Il componente si interfaccia con tutti i componenti che richiedono l'autenticazione.

Interfacce richieste:

- **Credenziali di accesso:** le credenziali includono email e password. Sono richieste all'utente per l'accesso alle funzioni del sito, ad eccezione della sola visualizzazione dello stato dei servizi e della classifica (i componenti per queste due ultime funzioni verranno descritti successivamente).
- **Dati account:** i dati dell'account, in tutti i ruoli, includono l'email e l'hash della password già salvati nel database, che verranno poi confrontati con le credenziali di accesso inserite.
- **Token:** necessario per identificare l'account dopo l'accesso con le credenziali e per verificare il ruolo dell'account, permettendo o meno all'utente di accedere alle varie funzionalità del sistema.

Interfacce offerte:

- **risposta autenticazione utente:** in base alle credenziali inserite, permette all'utente di accedere al servizio o meno. In particolare fornisce un token JWT che include il ruolo dell'account e altri dati di autenticazione (ad esempio l'id dell'account).
- **risposta autenticazione GdS:** in base alle credenziali inserite, permette al GdS di accedere al servizio o meno. In particolare fornisce un token JWT che include il ruolo dell'account e altri dati di autenticazione (ad esempio l'id dell'account).
- **risposta autenticazione admin:** in base alle credenziali inserite, permette al super admin di accedere al servizio o meno. In particolare fornisce un token JWT che include il ruolo dell'account e altri dati di autenticazione (ad esempio l'id dell'account).

CMP2: Gestione database

Descrizione:

il componente funziona da raccolta dati. Salva nel database le tutte le modifiche/creazioni relative a: account, servizi, sondaggi, segnalazioni.

Il componente si interfaccia con tutti i componenti che gestiscono i dati sopracitati.

Interfacce richieste:

- **dati database:** raccolta di tutti i dati presenti nel database.
- **dati nuovo utente:** dati relativi a un nuovo utente da aggiungere al database.
- **conferma richiesta:** modifica della richiesta di registrazione gds in modo che venga marcata come confermata.
- **dati nuovo feedback:** dati relativi a un nuovo feedback (compilazione del sondaggio) da aggiungere al database.
- **dati nuovo sondaggio:** dati relativi a un nuovo sondaggio da aggiungere al database.
- **dati nuovo servizio:** dati relativi a un nuovo servizio da aggiungere al database.
- **dati servizio modificato:** dati modificati di un servizio esistente.
- **dati nuova segnalazione:** dati relativi a una nuova segnalazione da aggiungere al database.
- **dati form segnalazione:** dati relativi a una form di segnalazione di un servizio, può essere sia creata che modificata.

Interfacce offerte:

- **modifiche database:** funzione che garantisce che tutti i dati modificati da qualsiasi interazione con altri componenti venga salvata correttamente nel database.
- **dati account:** fornisce i dati personali di qualsiasi account memorizzato nel database.
- **servizio associato:** fornisce il servizio associato a un GdS

CMP3: Gestione richieste di registrazione servizio

Descrizione:

Componente che si occupa di gestire la creazione e la conferma di una richiesta di registrazione di un servizio.

La richiesta include: username e password dell'utente gds e i dati del servizio che si vuole inserire (per la struttura vedere il componente "Gestione servizi")

Interagisce con i componenti:

- **Gestione autenticazione** per permettere agli account admin di confermare la richiesta di registrazione.
- **Gestione registrazione GdS** per creare un nuovo gds con i dati inseriti nella richiesta.
- **Gestione servizi** per creare un nuovo servizio con i dati inseriti nella richiesta.

Interfacce richieste:

- **dati e servizio GdS:** dati necessari alla compilazione della richiesta. Includono i dati relativi al servizio e al GdS da aggiungere.
- **autenticazione admin:** necessaria per la conferma della richiesta e la conseguente aggiunta nel database del nuovo GdS e admin.

Interfacce offerte:

- **Conferma richiesta:** dati della richiesta che si vuole confermare, in modo che venga marcata come tale nel database, questa funzione necessita dell'autenticazione con account dal ruolo admin.
- **dati servizio:** dati del servizio inserito, che sarà aggiunto al database dal componente "Gestione servizi" in seguito alla conferma della relativa richiesta.
- **nuovo GdS:** dati del GdS inserito, che sarà aggiunto al database dal componente "Gestione registrazione GdS" in seguito alla conferma della relativa richiesta.
- **lista delle richieste:** restituisce una lista con i dati di tutte le richieste di registrazione servizio presenti nel sistema, anche questa funzione necessita di dell'autenticazione come admin.

CMP4: Gestione registrazione GdS

Descrizione:

Il componente si occupa di gestire la creazione nel database di un GdS in seguito all'accettazione da parte di un admin della relativa richiesta.

Interagisce con:

- **Gestione richiesta di registrazione servizio** per avere i dati del GdS che deve aggiungere.
- **Gestione database:** per aggiungere al database il nuovo GdS.

Interfacce richieste:

- **nuovo GdS:** dati del nuovo GdS.

Interfacce offerte:

- **dati nuovo GdS:** dati del nuovo GdS da aggiungere al database.

CMP5: Gestione servizi

Descrizione:

Il componente si occupa della creazione, modifica e visualizzazione dei servizi.

Il servizio ha la seguente struttura: titolo, logo, descrizione, azienda, form di segnalazione, sito web.

Interagisce con:

- **Gestione database** per la modifica di un servizio.
- **Gestione richiesta registrazione servizio** per la creazione di un nuovo servizio.
- **Gestione autenticazione:** per la visualizzazione non è richiesta l'autenticazione, si potrà accedere alla visualizzazione di tutti i servizi presenti nel sistema e i loro

dettagli, mentre il ruolo “gds” potrà accedere solamente ai dettagli del proprio servizio.

La modifica invece è accessibile solo da “admin”, che può modificare tutti i servizi e “gds” che invece può modificare solo il proprio servizio.

Per quanto riguarda la creazione invece, essa avviene mediante l’accettazione della richiesta di registrazione, pertanto è attuabile solo dall’admin.

Interfacce richieste:

- **id servizio:** id del servizio di cui si vogliono visualizzare i dettagli.
- **modifiche servizio:** modifiche da effettuare al servizio, questi dati possono essere forniti solo da “admin” e “gds”.
- **elenco servizi:** dati riguardanti tutti i servizi presenti nel database del sistema.
- **nuovo servizio:** dati del nuovo servizio da aggiungere al sistema
- **autenticazione gds:** verifica che il gds sia autenticato per permettere di modificare il servizio.
- **autenticazione admin:** verifica che il super admin sia autenticato per permettere di modificare il servizio.

Interfacce offerte:

- **lista servizi:** visualizzazione di tutti i servizi presenti nel sistema.
- **dettaglio servizio:** visualizzazione dei dati relativi al singolo servizio il cui id è stato inserito in input in “id servizio”.
- **dati nuovo servizio:** dati del nuovo servizio da aggiungere al database, interfaccia usata da “Gestione database” per aggiungere il servizio al database.
- **dati servizio modificato:** dati nel servizio che è stato modificato, interfaccia usata da “Gestione database” per aggiornare i dati già presenti nel database.

CMP6: Gestione avvisi

Descrizione:

Il componente si occupa della creazione, e della visualizzazione degli avvisi.

Interagisce con:

- **Gestione database** per salvare i nuovi avvisi e per ottenere la lista dei servizi già esistenti.
- **Gestione autenticazione** per fornire ai vari tipi di utenti le interfacce a loro riservate. Gli utenti autenticati possono accedere alla visualizzazione degli avvisi, ma solo “gds” può crearne di nuovi per il suo servizio.
- **Filtro** per fornire all’utente la visualizzazione degli avvisi dei suoi servizi preferiti.

Interfacce richieste:

- **nuovo avviso:** dati riguardanti un avviso da aggiungere, interfaccia utilizzabile solo da un account di tipo ‘gds’.
- **id avviso:** id dell’avviso di cui si vogliono visualizzare i dettagli.
- **elenco avvisi:** l’elenco degli avvisi già presenti sul sistema, interfaccia utilizzata da “Gestione database”

- **autenticazione utente:** verifica che l'utente sia autenticato per permettergli di visualizzare gli avvisi mandati dai servizi.
- **autenticazione gds:** verifica che il gds sia autenticato per permettergli di inviare un nuovo avviso.

Interfacce offerte:

- **dati nuovo avviso:** dati del nuovo avviso da inserire nel database, interfaccia utilizzata dal database.
- **lista avviso:** lista degli avvisi presenti nel sistema.
- **dettagli avviso:** fornisce i dettagli di un avviso specifico. I dettagli includono: titolo, messaggio, data di inizio, data di fine.

CMP7: Gestione delle segnalazioni

Descrizione:

Il componente gestisce la creazione di una segnalazione, interagisce con:

- **Gestione autenticazione** per assicurarsi che l'account che manda una segnalazione sia autenticato.
- **Gestione della form di segnalazione del servizio** per avere la struttura della form.
- **Gestione classifica** per la visualizzazione della classifica.
- **Gestore dei grafici** per la visualizzazione delle segnalazioni
- **Gestione del database** per memorizzare la segnalazione.

Interfacce richieste:

- **nuova segnalazione:** dati relativi a una nuova segnalazione, solo un utente autenticato può mandare una segnalazione.
- **id servizio:** id del servizio del quale si vogliono visualizzare le segnalazioni.
- **form di segnalazione:** struttura della form di segnalazione per poterla visualizzare.
- **autenticazione utente:** si assicura che l'utente sia autenticato.

Interfacce offerte:

- **lista di segnalazioni:** lista di tutte le segnalazioni presenti nel sistema di un determinato servizio
- **dati nuova segnalazione:** dati nella segnalazione da aggiungere al database, interfaccia usata dal gestore del database.

CMP8: Gestione grafici

Descrizione:

Il componente si occupa di generare i grafici per la visualizzazione delle segnalazioni.

Interagisce con:

- **Gestione delle segnalazioni** per avere le segnalazioni memorizzate nel database e generare i relativi grafici.

Le funzionalità offerte da questo componente sono liberamente accessibili anche da utenti non autenticati.

Interfacce richieste:

- **lista segnalazioni**: lista di tutte le segnalazioni per un determinato servizio
- **id servizio**: id del servizio di cui si vogliono prelevare le segnalazioni

Interfacce offerte:

- **istogramma**: visualizzazione istogramma. I dati mostrati includono il numero di segnalazioni al giorno degli ultimi 10 giorni per un determinato servizio.
- **areogramma**: visualizzazione areogramma. I dati mostrati includono le motivazioni indicate nelle segnalazioni degli ultimi 2 giorni per un determinato servizio.

CMP9: Gestione classifica

Descrizione:

Questo componente si occupa solamente di visualizzare una classifica dei servizi in base al numero di segnalazioni ricevute.

Interagisce con:

- **Gestione segnalazioni** per poter contare le segnalazioni ricevute da ogni singolo servizio.
- **Gestione servizi** per poter avere una lista di tutti i servizi presenti.

Non è necessario essere autenticati per utilizzare questo componente.

Interfacce richieste:

- **lista servizi**: lista di tutti i servizi presenti nel sistema.
- **lista segnalazioni**: lista di tutte le segnalazioni ricevute da ciascun servizio ottenuto da "lista servizi"

Interfacce offerte:

- **classifica servizi**: visualizzazione della classifica dei servizi come indicato nella descrizione del componente.

CMP10: Gestione form di segnalazione:

Descrizione:

Questo componente si occupa di creare, modificare e visualizzare la form di segnalazione per ogni servizio. La form di segnalazione descrive le opzioni che l'utente può selezionare al momento della segnalazione. Per la creazione e modifica della form è necessaria l'autenticazione con ruolo "gds" o "admin", invece la visualizzazione è libera per tutti gli utenti che hanno eseguito l'autenticazione.

I componenti con cui interagisce sono:

- **Gestione delle segnalazioni** per fornire la form alle segnalazioni.
- **Gestione autenticazione** per assicurarsi che gli utenti siano autenticati.
- **Gestione database** per creare e modificare la form.

Interfacce richieste:

- **nuova form**: le opzioni da aggiungere alla form di segnalazione.
- **id servizio**: l'id del servizio di cui si vuole modificare la form.
- **autenticazione gds**: si assicura che il gds sia autenticato per permettergli di modificare la sua form.
- **autenticazione admin**: si assicura che l'admin sia autenticato per permettergli di modificare la form di un qualsiasi servizio.

Interfacce offerte:

- **form di segnalazione**: restituisce la form di segnalazione all'utente che vuole eseguire la segnalazione, e al database per aggiornarla.

CMP11: Gestione sondaggi:

Descrizione:

Il componente si occupa di creare e visualizzare la struttura di un sondaggio.

Il sondaggio può avere risposte singole, risposte multiple e risposte aperte.

Solo un account di tipo "gds" può creare e mandare un messaggio, mentre la visualizzazione è aperta solo agli utenti che hanno eseguito l'autenticazione.

Interagisce con:

- **Gestione database** per memorizzare il sondaggio inviato
- **Gestione autenticazione** per assicurarsi che gli utenti siano autenticati.

Interfacce richieste:

- **nuovo sondaggio**: dati relativi al sondaggio da creare nel database, solo gli utenti che hanno fatto l'accesso con il ruolo "gds" possono usare questa interfaccia.
- **id sondaggio**: id del sondaggio che di cui si vuole visualizzare la struttura.
- **id servizio**: id del servizio di cui si vogliono ottenere i sondaggi.

Interfacce offerte:

- **dati nuovo sondaggio**: i dati del nuovo sondaggio da inviare al database.
- **lista sondaggi**: restituisce la lista di sondaggi per singolo servizio.

CMP12: Gestione feedback:

Descrizione:

Quando un utente compila un sondaggio genera un feedback, che non è altro che un sondaggio compilato.

Il componente si occupa di creare e visualizzare i feedback.

Interagisce con:

- **Gestione database** per salvare i nuovi feedback sul database.
- **Gestione autenticazione** per assicurarsi che l'utente sia autenticato per poter mandare un feedback.

Interfacce richieste:

- **nuovo feedback**: dati di un nuovo feedback inviato da un utente da inserire nel database, solo gli account "utenti" autenticati possono accedere a questa funzione.
- **id sondaggio**: id di un sondaggio di cui si vogliono visualizzare i feedback, solo gli account "gds" autenticati possono accedere a questa funzione.
- **autenticazione utente**: necessaria per compilare un sondaggio e quindi per generare un feedback.
- **autenticazione gds**: necessaria per visualizzare i risultati di un sondaggio.

Interfacce offerte:

- **dati nuovo feedback**: dati di un nuovo feedback da inserire nel database, questa interfaccia verrà utilizzata da "Gestione database"
- **lista feedback**: restituisce la lista dei feedback per un singolo servizio, questa interfaccia può essere utilizzata solo con un account di tipo "gds"

CMP13: Gestione registrazione utente

Descrizione:

Questo componente si occupa di gestire la registrazione di un nuovo utente.

L'utente ha email e password.

Interagisce con:

- **Gestione database** per salvare nel database il nuovo utente

Interfacce richieste:

- **nuovo utente**: le credenziali relative al nuovo utente che si sta registrando.

Interfacce offerte:

- **dati nuovo utente**: i dati del nuovo utente da inserire nel database, interfaccia usata dal gestore del database.

CMP14: Gestione utente

Descrizione:

Il componente si occupa di gestire le informazioni base dell'utente, in particolare i servizi preferiti e le credenziali.

I dati dell'utente presentano inoltre un flag che distingue gli utenti "comuni" dai super admin.

Interagisce con:

- **Gestione database** per recuperare i dati dell'utente.
- **Gestione autenticazione** per assicurarsi che solo un utente autenticato possa accedere alle proprie informazioni riservate.
- **Filtro**, il componente fornisce infatti gli ID dei servizi preferiti che serviranno per filtrare.

Interfacce richieste:

- **autenticazione utente**: si assicura che l'utente sia autenticato per fornire i dettagli personali dell'utente.
- **dati account**: dati dell'account, incluse credenziali e IDs dei servizi preferiti, interfaccia utilizzata da "Gestione database".
- **id utente**: id dell'utente di cui si vogliono sapere le informazioni.

Interfacce offerte:

- **lista ID dei servizi preferiti**: elenco degli ID dei servizi preferiti dell'utente, richiede l'autenticazione.
- **dati utente**: restituisce le informazioni dell'utente autenticato, richiede l'autenticazione.

CMP15: Gestione GdS

Descrizione:

Il componente si occupa di fornire le informazioni di un servizio, in particolare il servizio associato.

Richiede l'autenticazione del GdS per visualizzare i dati.

Interagisce con:

- **Gestione database** per ottenere il servizio associato a un GdS.
- **Gestione autenticazione** per fornire i dati solo agli utenti autenticati correttamente.

Interfacce richieste

- **autenticazione gds**: si assicura che il gds sia autenticato per potergli fornire le informazioni.
- **ID gds**: ID del GdS di cui si vogliono ottenere le informazioni
- **servizio associato**: ottiene da "Gestione database" i dati del servizio associato all'ID del GdS inserito.

Interfacce offerte:

- **servizio**: servizio associato al GdS di cui è stato specificato l'ID

CMP16: Filtro dati utente

Descrizione:

Il componente serve per filtrare le raccolte di servizi, avvisi e sondaggi salvati nel database e mostrare solo quelle che interessano all'utente.

Di default verranno visualizzati i sondaggi e gli avvisi mandati dai servizi che gli utenti hanno tra i preferiti.

Inoltre nell'applicazione è presente un'opzione che permette di mostrare solamente i servizi preferiti, oltre ad una casella di testo che permette ricercare per nome i servizi.

Interagisce con:

- Tutti i componenti che gestiscono i dati che deve filtrare (avvisi, sondaggi, servizi).
- **Gestore autenticazione** per poter fornire all'utente i dati filtrati con le preferenze personali che ha impostato.
- **Gestione utente** per avere la lista di servizi preferiti.

Interfacce richieste:

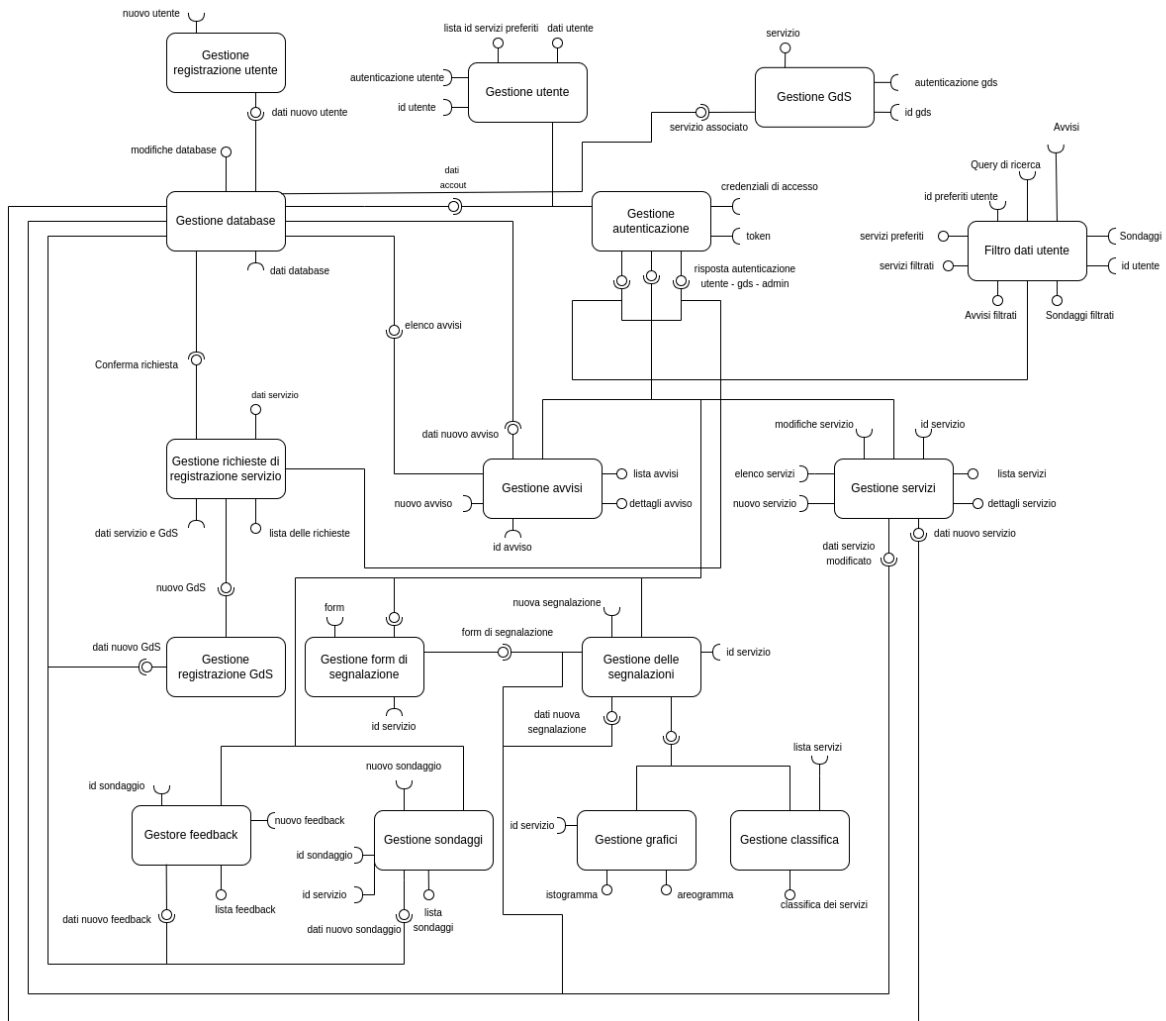
- **Query di ricerca:** input fornito dall'utente per ricercare un servizio specifico.
- **id preferiti utente:** IDs dei servizi preferiti dell'utente per applicare i filtri sui preferiti e mandare avvisi e sondaggi.
- **sondaggi:** lista dei sondaggi presenti nel sistema.
- **avvisi:** lista di avvisi presenti nel sistema.
- **id utente:** id dell'utente a cui bisogna far visualizzare i dati filtrati.

interfacce offerte:

- **servizi preferiti:** servizi filtrati sui preferiti dell'utente.
- **servizi filtrati:** servizi filtrati sulla query di ricerca.
- **avvisi filtrati:** avvisi mandati dai servizi preferiti dell'utente.
- **sondaggi filtrati:** sondaggi mandati dai servizi preferiti dell'utente.

2.2 Diagramma dei componenti

Viene riportato di seguito un diagramma complessivo di tutti i componenti di sistema e le loro interconnessioni descritte in precedenza.

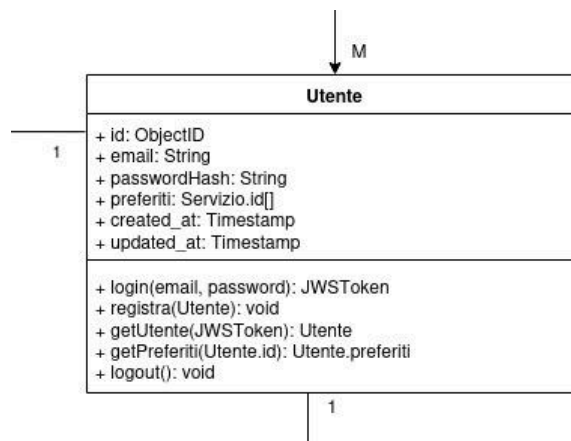


3. Diagramma delle classi

Nel presente capitolo vengono presentate le classi previste nell'ambito del progetto Trentus. Ogni componente presente nel diagramma dei componenti presentato nel capitolo precedente diventa una o più classi. Tutte le classi individuate sono caratterizzate da un nome, una lista di attributi che identificano i dati gestiti dalla classe e una lista di metodi che definiscono le operazioni previste all'interno della classe. Ogni classe può essere anche associata ad altre classi e, tramite questa associazione, è possibile fornire informazioni su come le classi si relazionano tra loro. Riportiamo di seguito le classi individuate, in questo processo si è proceduto anche nel massimizzare la coesione e minimizzare l'accoppiamento tra classi.

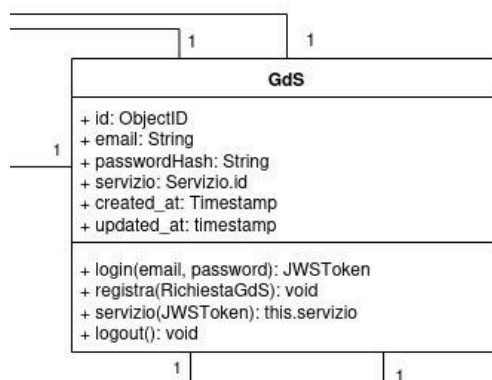
1. Utente

Abbiamo individuato l'attore Utente, cioè colui che visualizza i servizi e che può salvarli tra i preferiti. Per lui non sono necessarie molte informazioni se non l'email e la password per identificarli.



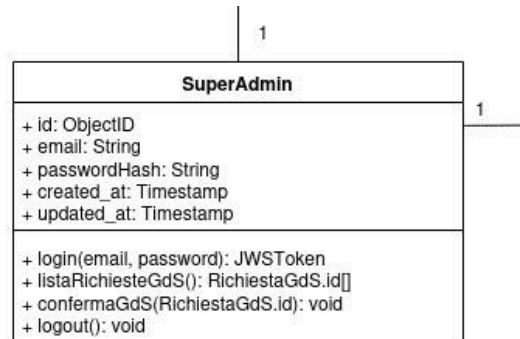
2. GdS

Abbiamo individuato l'attore GdS, cioè colui che gestisce il proprio servizio, il cui id è appunto salvato in *servizio*. Nemmeno lui necessita di molte informazioni se non le necessarie per identificarlo, cioè l'email e la password. Un'azione importante che può fare è la registrazione, compilando una *RichiestaGdS* per la creazione del suo Servizio da essere approvata



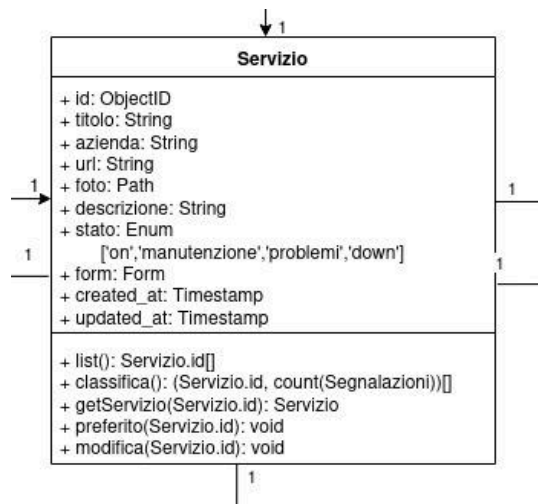
3. SuperAdmin

Il SuperAdmin è colui che gestisce la WebApp. Può accedere alle funzioni di tutte le classi identificate. Le funzioni uniche a se stesso sono però la lista e l'ammissione delle RichiestaGds per la creazione di un Servizio



4. Servizio

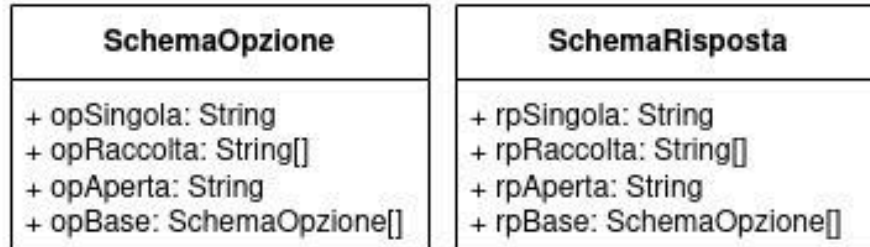
Per la rappresentazione di un Servizio è stata creata questa classe. Identifica in modo specifico tutto ciò che un utente può voler sapere riguardo al servizio, specialmente il suo *stato*, che può essere modificato. La funzione *preferito* serve all'utente per salvare/eliminare il servizio dai suoi preferiti, invece la funzione *classifica* serve per listare i servizi con più segnalazioni. Ogni servizio ha un form, che può essere modificato dal Gds ed è un documento che l'utente può compilare per rilasciare una segnalazione.



5. Schemi dei documenti

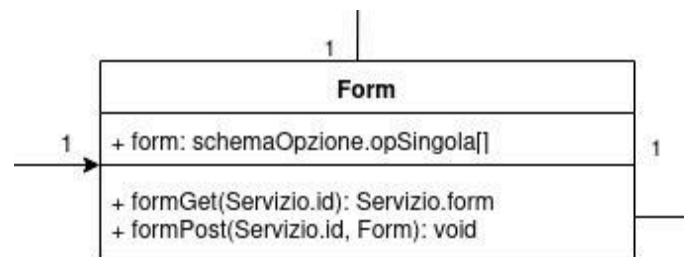
Servono per rappresentare i “questionari” compilabili / compilati dall'utente: schemaOpzione per i form ed i sondaggi, schema risposta per i feedback e le segnalazioni. La differenza tra i due è data dalla rappresentazione più che altro “logica” dei due elementi. Le differenze nel database sono minime, dovute solo al modo in cui sono salvate le scelte. Entrambi sono composti stringhe che rappresentano le intestazioni di opzioni spuntabili / spuntate dei questionari.

Queste possono essere singole (quindi domande true o false) o a raccolta (domande a scelta multipla) o aperte. *opBase* serve a rappresentare una collezione di scelte miste tra quelle sopra elencate.



6. Form

Il form è implementato come un array di dimensione variabile, impostata dal GdS secondo i suoi bisogni, di oggetti *schemaOpzione.opSingola*, che servono a rappresentare le domande che l'utente può spuntare. Ogni form può essere prelevato dall'utente e modificato dal GdS



7. Segnalazione

La segnalazione è il form compilato dall'utente, dove al posto di *schemaOpzione* si usa uno *schemaRisposta* che serve ad indicare se l'utente ha selezionato la domanda o no.

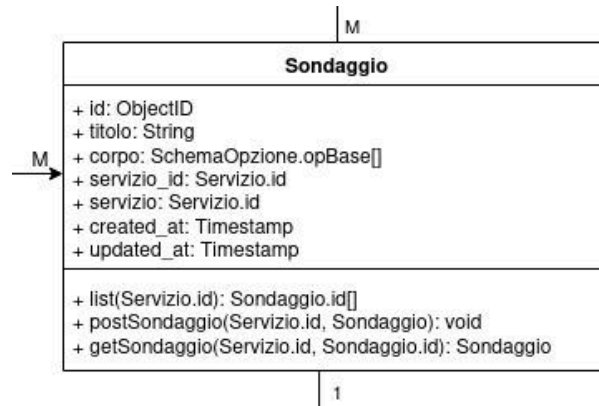
La segnalazione può essere compilata dall'utente dopo aver prelevato il form. Il GdS può facilmente elaborare i dati di multiple segnalazioni, esportando solo i commenti rilasciati dall'utente sotto questi, oppure visualizzando l'andamento / la frequenza delle opzioni selezionate tramite la visualizzazione di questi in appositi grafici



8. Sondaggio

Il GdS può rilasciare multipli questionari appositi che l'utente può compilare per far sapere la sua opinione riguardo al servizio, oppure altre necessità che il GdS può avere, magari domande sull'implementazione di una nuova funzionalità nel servizio.

Questi possono essere rappresentati in modo più diversificato, potendo sfruttare tutti i tipi di schemaOpzione disponibili



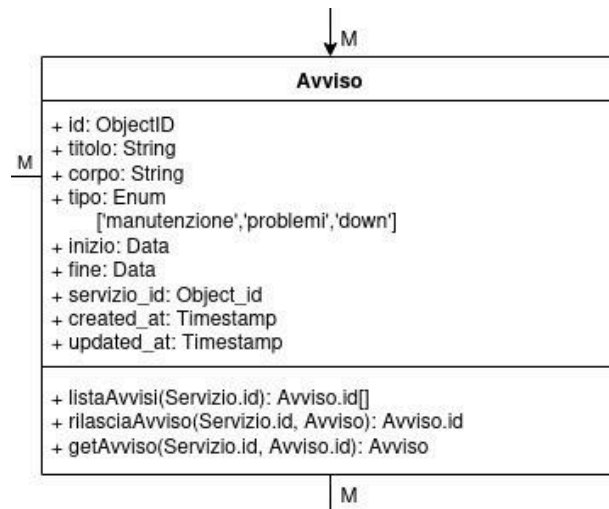
9. Feedback

Questi sono la rappresentazione compilata di un Sondaggio, dove ogni oggetto schemaOpzione viene convertito in un oggetto schemaRisposta in base alla scelta dell'Utente. Questi possono essere compilati dopo che l'utente ha ottenuto il sondaggio. Possono poi essere visualizzati dal GdS tramite la funzione risultati



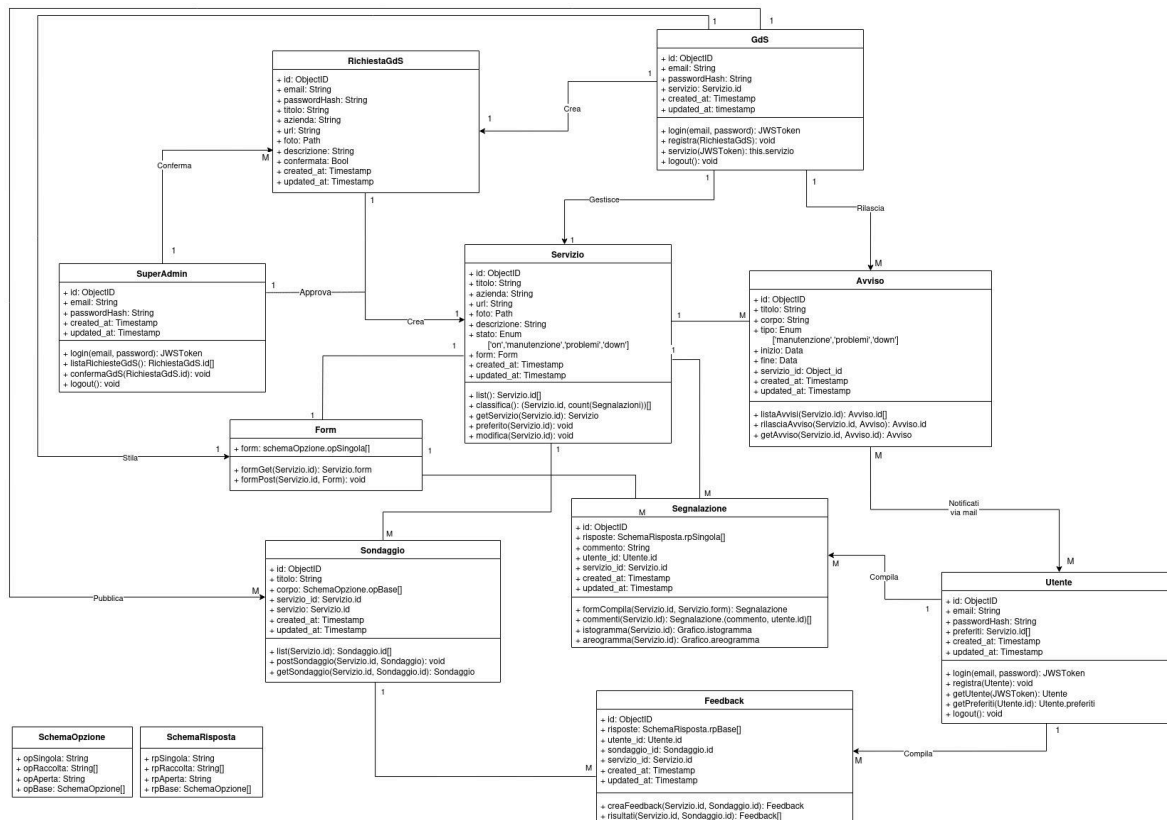
10. Avvisi

Per notificare agli utenti quando un servizio ha un problema esiste l'Avviso. Questo ha un titolo, il corpo, cioè la motivazione, ed il tipo di problema che il servizio ha, che corrisponde allo stato di questo. Ha poi una data di inizio ed una di fine per avvisare l'utente quando il servizio sarà nuovamente disponibile. Nota bene: la funzione rilascia avviso gestisce anche l'invio della mail agli utenti che hanno il servizio nei preferiti.



3.1 Diagramma delle classi

Riportiamo di seguito il diagramma delle classi con tutte le classi fino ad ora presentate. Qui si accentua graficamente la relazione tra i diversi attori ed i diversi oggetti di Trentus.



4. Dal class diagram alle API

Riportiamo di seguito una tabella con la mappatura tra i metodi presenti nel diagramma delle classi del capitolo precedente e le API individuate nel documento D3. Tutte le request e le response (tranne una piccola eccezione) sono gestite in JSON. Varie request si basano su dati contenuti nel token. Se il token non è presente apparirà il messaggio { "message": "Token non valido" }

| Classe | Metodo | HTTP Meth. | URI + query params | Request | Response | Note |
|--------|----------------|------------|---------------------------------------|---|---|--|
| Utente | getUtente() | GET | /utente | <i>Authorization field in headers has to contain the user's token</i> | { "_id": "6793f736...", "email": "Utente@email.com", "preferiti": [] } | L'array preferiti è popolato dai preferiti dell'utente |
| | login() | POST | /utente/login | { "email": "Utente@email.com", "password": "Password" } | { "message": "Login effettuato", "token": "eyJhbGciOiJIUzI..." } oppure { "message": "Dato* non corretto" } | *email o password |
| | registra() | POST | /utente/registrazione | { "email": "Utente@email.com", "password": "Password" } | { "message": "Utente registrato con successo" } oppure { "message": "Email già registrata" } | |
| | getPreferiti() | GET | /utente/preferiti | <i>Authorization field in headers has to contain the user's token</i> | [{ "_id": "67950...", ... }] | |

| | | | | | | |
|-------------------|---------------------|------|---|--|--|--|
| GdS | login() | POST | /gds/login | { "email": "GdS@email.com", "password": "Password" } | { "message": "Login effettuato", "token": "aygkbPciPxJBWzz..." } oppure { "message": "GdS inesistente" } | |
| | registra() | POST | /gds/registrazione | data: { "email": "GdS@email.com", "password": "Prova", "titolo": "Servizio", "azienda": "Azienda", "url": "azienda.com", "descrizione": "GdS Test" } foto: (logo del servizio) | { "message": "Richiesta inviata con successo" } oppure { "message": "Richiesta già spedita" } | NB: questa è una delle due API che non richiede la request in JSON bensì in <i>multipart/form-data</i> |
| | servizio() | GET | /gds/servizio | <i>Authorization field in headers has to contain the gds's token</i> | { "servizio_id": "67950e7..." } | |
| SuperAdmin | login() | POST | /controlpanel/login | { "email": "Utente@email.com", "password": "Password" } | { "message": "Login effettuato", "token": "eyJhbGciOiJIUzI..." } oppure { "message": "Dato* non corretto" } | Nel caso un utente non SuperAdmin tenti di accedere verrà mostrato un messaggio di errore |
| | listaRichiesteGdS() | GET | /controlpanel/conferma_gds/lista | | [{ Richiesta1GdS* }, { Richiesta2GdS* }, ...] | *si intende l'oggetto, che per esteso occuperebbe troppo spazio |
| | confermaGdS() | POST | /controlpanel/conferma_gds/{richiesta_id} | | { "message": "GdS creato con successo" } oppure * { "message": "Richiesta non trovata" } | *anche per richiesta già confermata |

| | | | | | | |
|----------|---------------|------|---|--|---|--|
| Servizio | list() | GET | /servizi/ | | [{ "_id": "9fb775...", ... }] | |
| | classifica() | GET | /servizi/classifica | | [{ "segnalazioni": 3, "servizio_id": "e9fb775..." } ...] | |
| | getServizio() | GET | /servizi/{servizio_id} | | { "_id": "67a9ca1...", "titolo": "Servizio", "azienda": "Azienda", "url": "azienda.com", "foto": "/images/f6a03...", "descrizione": "GdS Test", "stato": "on", "form": [], "createdAt": "2025-02-10T09", "updatedAt": "2025-02-10T09", "__v": 0 } | |
| | preferito() | POST | /servizi/{servizio_id}/preferito | Authorization field in headers has to contain the user's token | { "message": "Servizio eliminato" } oppure { "message": "Servizio salvato" } | |
| | modifica() | POST | /servizi/{servizio_id}/modifica | Authorization field in headers has to contain the gds's token data: { "titolo": "Servizio", "azienda": "Azienda", "url": "azienda.com", "descrizione": "GdS Test" "stato": "on" } foto: (logo del servizio) | { "message": "Servizio aggiornato con successo", Dati aggiornati } oppure { "message": "Errore nella modifica del servizio" } | |

| | | | | | | |
|---------------------|--------------|------|---|--|--|---|
| Form | formGet() | GET | /servizi/{servizio_id}/segnalazioni/form | Authorization field in headers has to contain the user's token or the gds's token | { "_id": "67a9ca1...", "form": ["Prob1", "Prob2", "Prob3"] } | |
| | formPost() | POST | /servizi/{servizio_id}/segnalazioni/form | Authorization field in headers has to contain the gds's token { "opzioni": ["Prob1", "Prob2", "Prob3"] } | { "message": "Form creata con successo" } | Ogni servizio può avere una sola form, non ci sono problemi di duplicazione |
| Segnalazioni | compila() | POST | /servizi/{servizio_id}/segnalazioni/compila | Authorization field in headers has to contain the user's token { "risposte": ["Prob1"], "commento": "Prob1 is blah blah" } | { "message": "Segnalazione creata con successo" } oppure { "message": "Hai già compilato una segnalazione oggi." } | L'array risposte o il commento possono essere vuoti, ma non entrambi |
| | commenti() | GET | /servizi/{servizio_id}/segnalazioni/commenti | | [{ "_id": "67a9d4fb...", "commento": "Prob1 is blah blah", "utente_id": { "_id": "67a9ce3...", "email": "Utente@email.com" } }, ...] | |
| | istogramma() | GET | /servizi/{servizio_id}/segnalazioni/istogramma | | [{ "date": "2-10", "count": 3 }, ...] | |
| | areogramma() | GET | /servizi/{servizio_id}/segnalazioni/areogramma | | [{ "risposta": { "risposta": "Prob1" }, "count": 1, "percentage": 100 }, ...] | |

| | | | | | | |
|-----------------|-----------------|------|---|---|--|--|
| Sondaggi | list() | GET | /servizi/{servizio_id}/sondaggi/ | Authorization field in headers has to contain the user's token or the gds's token | [{ "_id": "wi4875..." }, ...] | |
| | sondaggioPost() | POST | /servizi/{servizio_id}/sondaggi/ | Authorization field in headers has to contain the gds's token <pre>{ "titolo": "Quality Survey", "opzioni": [{ "type": "singola", "opzione": "Buono" }, ... { "type": "raccolta", "domanda": "Colore", "opzioni": [{ "opzione": "Rosso" }, { "opzione": "Verde" }, { "opzione": "Blu" }], "tipo": "checkbox" }, { "type": "aperta", "domanda": "Dicci la tua" }] }</pre> | { "message": "Sondaggio creato con successo" } | |
| | sondaggioGet() | GET | /servizi/{servizio_id}/sondaggi/{sondaggio_id} | Authorization field in headers has to contain the user's token or the gds's token | { "_id": "67a9dc32...", "titolo": "Quality Survey", "corpo": [...] } | |

| | | | | | | |
|-----------------|----------------|------|---|--|--|--|
| Feedback | creaFeedback() | POST | /servizi/{servizio_id}/sondaggi/{sondaggio_id} | <pre>{ "risposte": [{ "type": "singola", "risposta": "Luca" }, { "type": "raccolta", "domanda": "Ciao", "risposte": [{ "risposta": "Rosso" }], "tipo": "checkbox" }, { "type": "aperta", "domanda": "Domanda", "risposta": "Risposta" }] }</pre> | <pre>{ "message": "feedback rilasciato con successo" } oppure { "message": "Hai già compilato questo sondaggio" }</pre> | |
| | risultati() | GET | /servizi/{servizio_id}/sondaggi/{sondaggio_id}/risultati | <p>Authorization field in headers has to contain the gds's token</p> | <pre>[{ "_id": "67a9e1a...", "risposte": [...], "utente_id": "67a9ce...", "sondaggio_id": "67a9d...", "servizio_id": "67a9ca1...", "createdAt": "2025-02-10T...", "updatedAt": "2025-02-10T...", "__v": 0 }]</pre> | |

| | | | | | | |
|---------------|------------------|------|---|---|---|--|
| Avvisi | listaAvvisi() | GET | /servizi/{servizio_id}/avvisi/ | | [{ "_id": "ati4r75...", ... }] | |
| | rilasciaAvviso() | POST | /servizi/{servizio_id}/avvisi/ | Authorization field in headers has to contain the gds's token { "titolo": "Avviso", "corpo": "Contenuto dell'avviso", "tipo": "manutenzione", "inizio": "2025-12-03", "fine": "2026-12-16" } | "error": "alert validation failed: inizio: La data di inizio non deve essere nel passato" oppure "error": "alert validation failed: fine: La data di fine deve essere successiva alla data di inizio." oppure Avviso creato e email inviata! | *modificare le date per vedere i diversi risultati |
| | getAvviso() | GET | /servizi/{servizio_id}/avvisi/{avviso_id} | | [{ "_id": "67a9e621...", "titolo": "Avviso", "corpo": "Contenuto dell'avviso", "tipo": "manutenzione", "inizio": "2025-12-03T", "fine": "2026-12-16T", "servizio_id": "67a9ca1...", "createdAt": "2025-02-10T", "updatedAt": "2025-02-10T", "__v": 0 }] | |