

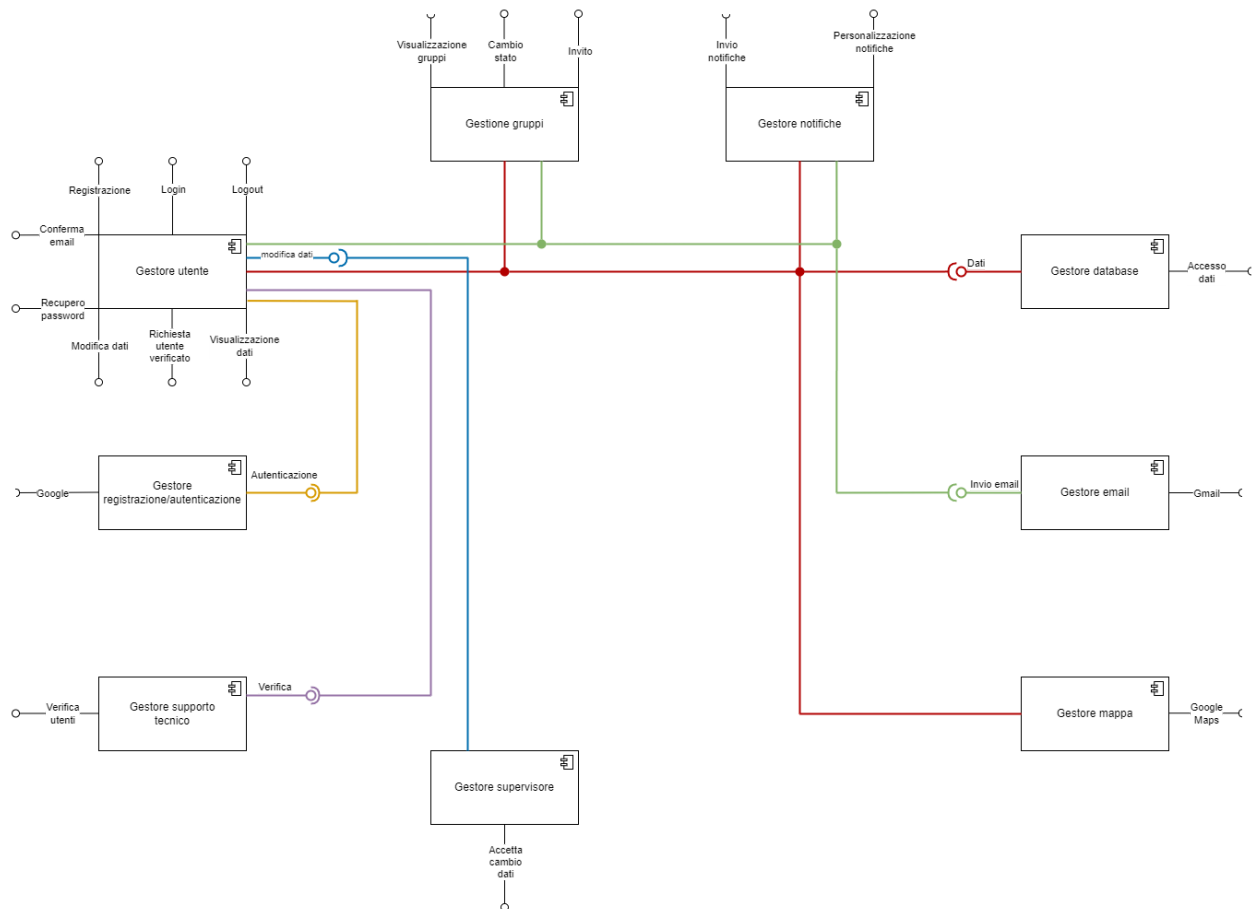
Deliverable Two - Gruppo #6

Pedrini Andrea - 226607

Martinico Marco - 230509

Santus Tommaso - 226666

Components diagram



Analisi interfacce

- Gestore utente:

Questo componente gestisce la registrazione di utenti anonimi al sistema, il login, logout e gestione del proprio account agli utenti registrati.

TIPOLOGIA	NOME	DESCRIZIONE
Fornita	Registrazione	Il componente fornisce un'interfaccia all'utente per registrarsi (RF1.5)
Fornita	Login	Il componente fornisce un'interfaccia all'utente per effettuare il login (RF1.7)
Fornita	Logout	Il componente fornisce un'interfaccia all'utente per effettuare il logout (RF2.3)
Fornita	Conferma email	Il componente permette all'utente di confermare la propria email collaborando con il gestore notifiche
Fornita	Recupero password	Il componente fornisce un'interfaccia per recuperare la password tramite notifiche email
Fornita	Visualizzazione dati	Il componente fornisce un'interfaccia all'utente per visualizzare i propri dati sensibili
Fornita	Richiesta utente verificato	Il componente fornisce un'interfaccia all'utente per richiedere l'upgrade a utente verificato
Fornita	Modifica dati	Il componente fornisce un'interfaccia in cui l'utente può modificare determinati dati. In caso l'utente sia verificato la modifica

		richiederà l'intervento del supervisore
--	--	---

- Gestore registrazione/autenticazione:

Questo componente gestisce l'autenticazione dell'utente che viene effettuata tramite un token inviato ad ogni richiesta effettuata dall'utente.

TIPOLOGIA	NOME	DESCRIZIONE
Fornita	Autenticazione	Il componente fornisce un'interfaccia per autenticarsi
Richiesta	Google	Il componente fornisce un'interfaccia per autenticarsi

- Gestore supporto tecnico:

Questo componente gestisce la verifica degli utenti che ne fanno richiesta.

TIPOLOGIA	NOME	DESCRIZIONE
Fornita	Verifica utenti	Il componente fornisce un'interfaccia al supporto tecnico per verificare gli utenti

- Gestore supervisore:

Questo componente gestisce il cambio dati degli utenti.

TIPOLOGIA	NOME	DESCRIZIONE
Fornita	Accetta cambio dati	Il componente permette di confermare il cambio dati

- Gestione gruppi:

Questo componente gestisce la visualizzazione dei gruppi e i relativi inviti. Inoltre gestisce anche i cambi di stato dei rifiuti immessi dall'utente.

TIPOLOGIA	NOME	DESCRIZIONE
Fornita	Invito	Il componente fornisce un'interfaccia per invitare altri utenti all'interno del proprio gruppo
Fornita	Cambio stato	Il componente fornisce un'interfaccia per modificare lo stato dei rifiuti all'interno del gruppo
Richiesta	Visualizzazione gruppi	Il componente fornisce un'interfaccia per visualizzare ogni informazione relativa al gruppo selezionato, tra cui la classifica di raccolta rifiuti tra i partecipanti

- Gestore notifiche:

Questo componente gestisce le notifiche selezionate dall'utente per da filtrarle, in modo che l'utente riceva email che riguardano solo le notifiche selezionate.

TIPOLOGIA	NOME	DESCRIZIONE
Fornita	Personalizzazione notifiche	Il componente fornisce un'interfaccia per selezionare le notifiche che si desiderano ricevere
Richiesta	Invio notifiche	Il componente invia le notifiche collaborando con il gestore email che notificherà l'utente

- Gestore database:

Questo componente permette agli altri componenti di accedere ai dati del database.

TIPOLOGIA	NOME	DESCRIZIONE
Fornita	Dati	Il componente fornisce un'interfaccia per accedere ai dati salvati sul database
Richiesta	Accesso dati	Il componente utilizza un'interfaccia fornita dal database per accedere ai dati salvati sul database

- Gestore email:

Questo componente gestisce le interazioni con il sistema esterno Gmail.

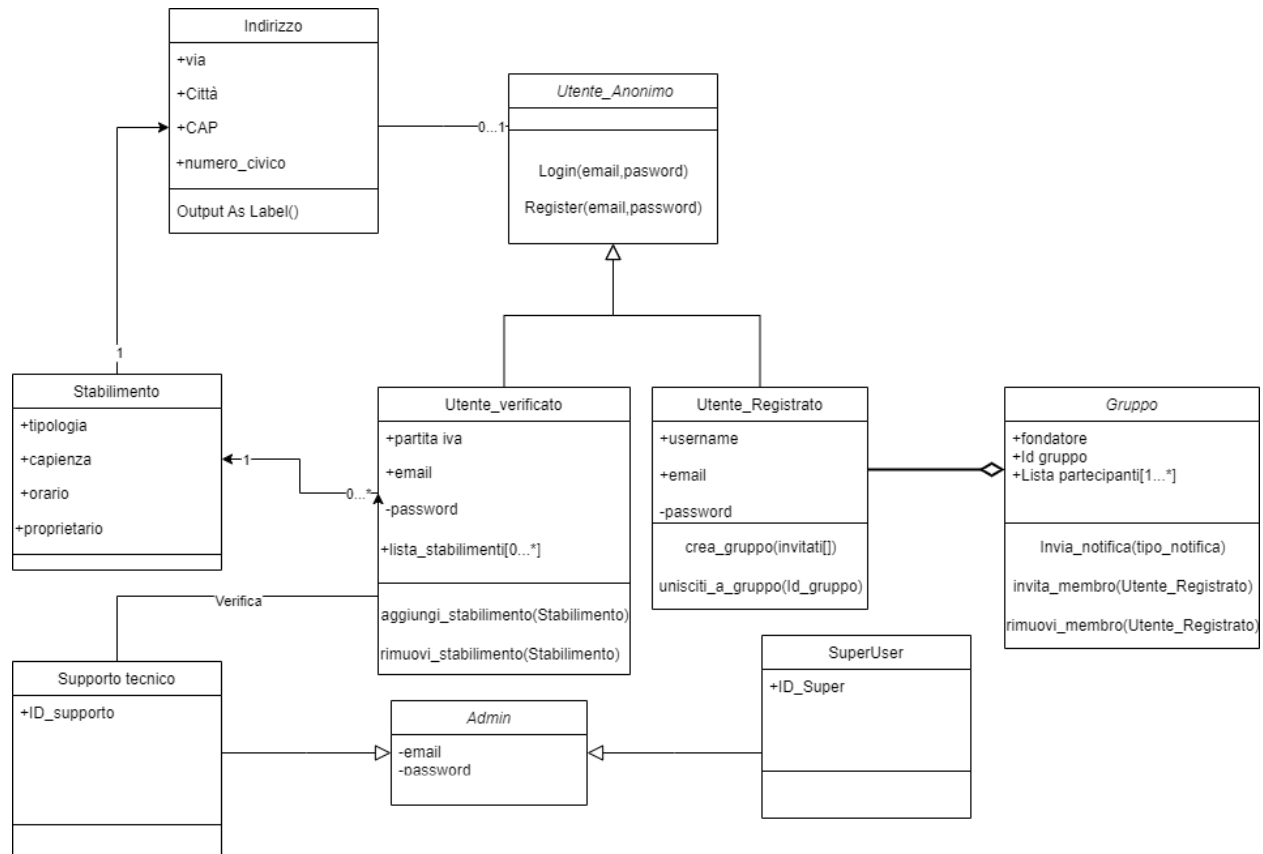
TIPOLOGIA	NOME	DESCRIZIONE
Richiesta	Gmail	Il componente utilizza l'interfaccia fornita da Gmail per richiedere l'invio di email

- Gestore mappa:

Questo componente gestisce le interazioni con il sistema esterno Google Maps.

TIPOLOGIA	NOME	DESCRIZIONE
Richiesta	Google Maps	Il componente utilizza l'interfaccia fornita da Google Maps per recuperare la mappa desiderata

Class diagram



Utente_Anonimo è la classe utente base da cui derivano Utente_registrato e Utente_verificato

i metodi di Utente anonimo sono Login e Register che consentono ad utente anonimo di diventare Utente_Registrato o Utente_verificato.

Metodo	Descrizione
Register()	se i dati inseriti sono nel formato corretto e la mail non è già presente nel database crea una nuova istanza di utente_Registrato
Login()	controlla il database e trasforma l'istanza di utente anonimo in un Utente_Registrato

la classe Utente_Registrato possiede 3 attributi base: username email e password. I suoi metodi crea_Gruppo e unisciti_a_gruppo gli permettono di interagire con la

classe Gruppo.

Metodo	Descrizione
crea_gruppo()	crea un nuovo gruppo e invia un invito ad ogni utente presente nella lista degli invitati fornita
unisciti_a_gruppo()	se provvisto di un invito associa l'utente Registrato al gruppo invitante

Gruppo è una classe data dall'aggregazione di più Utente_Registrato essa ha un fondatore un Id e una lista di partecipanti, i suoi metodi permettono interazioni con i membri del gruppo quali l'invio di notifiche e la aggiunta/rimozione di membri.

Metodo	Descrizione
Invia_notifica()	invia una notifica ad ogni membro del gruppo
invita_membro()	invia un invito ad un Utente_Registrato per potersi unire al gruppo
rimuovi_membro()	rimuovi un Utente_Registrato da lista_partecipanti

la classe Utente_verificato possiede gli attributi partita iva, email, password, e lista_stabilimenti i suoi metodi includono aggiungi stabilimento e rimuovi stabilimento che permettono appunto di modificare la lista di stabilimenti posseduti da un utente verificato.

Metodo	Descrizione
aggiungi_stabilimento()	se i dati inseriti sono corretti crea un nuovo stabilimento e lo inserisce in lista_stabilimenti
rimuovi_stabilimento()	rimuove uno stabilimento da lista_stabilimenti

la classe Stabilimento ha come attributi tipologia capienza orario e proprietario, la classe stabilimento si interfacerà con le funzioni della mappa interattiva.

Constraint OCL

- Classe Stabilimento

Descrizione: Uno stabilimento deve avere uno ed un solo utente verificato
context Stabilimento inv: self.proprietario <> null

Descrizione: l'indirizzo deve possedere un CAP di Trento

context Stabilimento

inv: self.Indirizzo.CAP ≥ 38121 and self.Indirizzo.CAP ≤ 38123

- Classe Utente

Descrizione: La mail di un utente non anonimo deve essere inserita

context Utente

inv:

self.email <> null

Descrizione: La password di un utente non anonimo deve contenere almeno 8 caratteri e meno di 32

context Utente::register(email, password)

pre: password.size() ≥ 8 and password.size() ≤ 32

- Classe Utente_verificato

Descrizione: aggiungi_Stabilimento aggiunge un nuovo stabilimento

context Utente_verificato::aggiungi_Stabilimento (Stabilimento)

post: self.Lista_stabilimenti@pre \rightarrow including (Stabilimento)

Descrizione: rimuovi_Stabilimento rimuove uno stabilimento

context Utente_verificato::rimuovi_Stabilimento (Stabilimento)

post: self.Lista_stabilimenti@pre \rightarrow excluding (Stabilimento)

- Classe Utente_Registrato

Descrizione: crea_gruppo data una lista di invitati crea un nuovo gruppo

context Utente_Registrato::crea_gruppo(invitati[])

post: self.Gruppo.fondatore = self.username

Descrizione: unisciti_a_gruppo accettando l'invito, l'utente registrato si unirà al gruppo

context Utente_Registrato::unisciti_a_gruppo(id_gruppo)

post: self.Gruppo.Lista_partecipanti@pre \rightarrow including (Utente_Registrato)

- Classe Gruppo

Descrizione: Un gruppo deve includere almeno 2 membri

context Gruppo

inv: self.Lista_partecipanti → size() ≥ 2

Descrizione: invita_membro aggiunge un nuovo membro

context Gruppo:: invita_membro (Utente_Registrato)

post: self.Lista_partecipanti@pre → including (Utente_Registrato)

Descrizione: rimuovi_membro rimuove un membro

context Gruppo:: rimuovi_membro (Utente_Registrato)

post: self.Lista_partecipanti@pre → excluding (Utente_Registrato)