



SocialBridge

System Design Document

SocialBridge

Riferimento	C18_SDD
Versione	0.5
Data	23/10/2024
Destinatario	Prof.ssa Filomena Ferrucci, Prof. Fabio Palomba
Presentato da	C18 team
Approvato da	

Team Members

<i>Nome</i>	<i>Cognome</i>	<i>Ruolo</i>	<i>Acronimo</i>	<i>Contatto</i>
Francesco	Peluso	PM	FP	f.peluso25@studenti.unisa.it
Luciano	Bercini	PM	LB	l.bercini@studenti.unisa.it
Mariarosaria	Rossi	PM	MR	m.rossi60@studenti.unisa.it
Fabio	Di Lallo	TM	FDL	f.dilallo3@studenti.unisa.it
Michele	Di Meo	TM	MDM	m.dimeo7@studenti.unisa.it
Luca	Fiore	TM	LF	l.fiore20@studenti.unisa.it
Andrea	Senatore	TM	AS	a.senatore158@studenti.unisa.it
Francesco	De Felice	TM	FDF	f.defelice4@studenti.unisa.it
Matteo	Nocerino	TM	MN	m.nocerino8@studenti.unisa.it
Alfio	Marra	TM	AM	a.marra39@studenti.unisa.it

Revision History

<i>Data</i>	<i>Versione</i>	<i>Descrizione</i>	<i>Autori</i>
06/11/2024	0.1	Prima Stesura	FDL, MDM, LF, AS, FDF, MN, AM
06/11/2024	0.2	Stesura paragrafi 1.1, 1.2, 1.3, 1.4, 1.5, 2	FDL, MDM, LF, AS, FDF, MN, AM
08/11/2024	0.3	Stesura paragrafi 3, 3.1, 3.2, 3.3, 3.4, 3.5	FDL, MDM, LF, AS, FDF, MN, AM
11/11/2024	0.4	Stesura paragrafi 3.5, 3.5.1, 3.6, 3.7, 4, 5	FDL, MDM, LF, AS, FDF, MN, AM
14/11/2024	0.5	Revisione Finale	FDL, MDM, LF, AS, FDF, MN, AM

Sommario

●	<i>Team Members</i>	2
●	<i>Revision History</i>	2
●	<i>1: Introduzione</i>	4
○	<i>1.1: Obiettivi del Sistema</i>	4
○	<i>1.2: Design Goal</i>	4
○	<i>1.2.1: Design Goal</i>	4
○	<i>1.2.2: Trade-offs</i>	6
○	<i>1.3: Definizioni, acronimi e abbreviazioni</i>	8
○	<i>1.4: Riferimenti</i>	8
○	<i>1.5: Overview</i>	8
●	<i>2: Architettura del Sistema Attuale</i>	8
●	<i>3: Architettura del Sistema Proposto</i>	9
○	<i>3.1: Sintesi del Sistema</i>	9
○	<i>3.2: Decomposizione in sottosistemi</i>	9
○	<i>3.2.1: Diagramma delle componenti</i>	10
○	<i>3.2.2: Diagramma architetturale dei sottosistemi</i>	11
○	<i>3.3: Mapping Hardware e Software</i>	12
○	<i>3.4: Gestione dei dati persistenti</i>	12
○	<i>3.5: Sicurezza ed Accessi</i>	17
○	<i>3.5.1: Matrice degli accessi</i>	18
○	<i>3.6: Controllo globale del software</i>	19
○	<i>3.7: Condizioni limite</i>	20
●	<i>4: Servizi dei sottosistemi</i>	23
●	<i>5: Glossario</i>	25

1. Introduzione

1.1: Obiettivi del sistema

SocialBridge è una piattaforma web inclusiva progettata per facilitare la partecipazione a eventi sociali per persone introversive o con disabilità. Gli obiettivi principali del sistema sono:

Consentire agli utenti di trovare eventi sociali che rispettino le loro esigenze personali.

Facilitare la creazione di una comunità basata su interessi comuni, migliorando il senso di appartenenza e promuovendo legami significativi tra gli utenti.

Assicurare un ambiente sicuro e accogliente, con particolare attenzione all'accessibilità e alla personalizzazione.

1.2: Design Goal

In questa sezione sono specificati tutti i design goal che il sistema punta a raggiungere. Ogni design goal è caratterizzato da:

- **Rank**
- **ID Design Goal**
- **Descrizione**
- **Categoria** (Performance, Dependability, Cost, Maintenance, End user)
- **Origine** (ID del RNF associato)

Poiché è impossibile realizzare una piattaforma "perfetta" e priva di errori, in questa sezione sono elencati anche dei trade-off che rappresentano compromessi necessari quando si prendono decisioni che coinvolgono la gestione delle risorse limitate o quando si cerca di massimizzare i benefici in situazioni dove non è possibile soddisfare completamente tutti gli obiettivi contemporaneamente.

1.2.1: Design Goal

Rank	ID	Descrizione	Categoria	Origine
1	DG_1	Sicurezza dei Dati Personali: Solo l'utente autenticato può modificare i propri dati personali. Il sistema deve fornire un feedback visibile entro 2 secondi.	Sicurezza	RNF_1
2	DG_2	Accessibilità e Inclusione: Garantire che il sistema fornisca informazioni chiare sull'accessibilità degli eventi, rispettando requisiti base di leggibilità.	Accessibilità	RNF_2
3	DG_3	Usabilità e Chiarezza Interattiva: Offrire conferme immediate per azioni come salvataggi e prenotazioni eventi, migliorando la comunicazione con messaggi di errore chiari.	Usabilità	RNF_3, RNF_6
4	DG_4	Esperienza Personalizzata e Accuratezza nelle Valutazioni: Fornire un sistema di raccomandazioni e affinità tra utenti con almeno il 90% di accuratezza, basato su interessi e attività comuni.	Esperienza utente	RF_7, RNF_4
5	DG_5	Efficienza del Motore di Ricerca: Il sistema deve rispondere alle ricerche di eventi e utenti entro 2 secondi, anche su database di grandi dimensioni.	Performance	RNF_5
6	DG_6	Feedback e Notifiche Tempestive: Inviare conferme e notifiche entro pochi secondi per azioni cruciali come prenotazioni, recensioni e modifiche ai dati.	Notifiche	RF_10, RF_12, RNF_6
7	DG_7	Gestione delle Segnalazioni e Moderazione: Fornire un sistema semplice per segnalare contenuti o comportamenti non appropriati, con risposta simulata entro 48 ore.	Sicurezza	RF_9
8	DG_8	Chat Privata Accessibile e Sicura: Fornire una chat privata tra utenti che abbiano partecipato a eventi comuni, con opzioni di segnalazione e archiviazione.	Comunicazione	RF_14
9	DG_9	Chiarezza del Chatbot e Accessibilità per il Supporto: Garantire risposte semplici e accessibili tramite chatbot per utenti con competenze digitali diverse.	Usabilità	RNF_7
10	DG_10	Conformità GDPR e Trasparenza: La piattaforma deve rispettare pienamente le normative GDPR, fornendo trasparenza nel trattamento dei dati e opzioni di gestione delle informazioni personali.	Compliance	RNF_2

1.2.1: Trade-offs

Trade-offs	Descrizione
Sicurezza dei Dati vs. Facilità d'Uso	<p>L'implementazione di sistemi di sicurezza avanzati, come la verifica email e l'autenticazione per modifiche ai dati personali, può rendere i processi di accesso e modifica meno immediati per gli utenti.</p> <p>SocialBridge darà priorità alla sicurezza dei dati personali, con notifiche e conferme immediate, pur cercando di mantenere il flusso intuitivo per l'utente.</p>
Accessibilità vs. Complessità del Design	<p>Integrare elementi di accessibilità (come opzioni per eventi inclusivi e supporto per screen reader) può aumentare la complessità del design e i costi di sviluppo.</p> <p>Prioritizzare l'accessibilità seguendo le linee guida WCAG e mantenendo una struttura intuitiva e coerente. L'inclusività è fondamentale per gli obiettivi del progetto, quindi si investirà nelle funzionalità chiave per una UX inclusiva.</p>
Performance vs. Precisione delle Raccomandazioni	<p>Il calcolo accurato dell'affinità tra utenti per personalizzare l'esperienza richiede analisi approfondite, che possono impattare sui tempi di risposta.</p> <p>Implementare algoritmi di raccomandazione progressivi, ottimizzando per accuratezza e velocità. Verranno bilanciati i livelli di personalizzazione con prestazioni elevate per garantire una UX fluida.</p>
Efficienza del Motore di Ricerca vs. Scalabilità del Database	<p>Fornire risultati di ricerca rapidi (< 2 secondi) con un database di grandi dimensioni può richiedere risorse aggiuntive per infrastrutture avanzate.</p> <p>Utilizzare tecnologie ottimizzate per le ricerche su database estesi, implementando una struttura scalabile e adottando caching dove possibile, bilanciando i costi.</p>
Notifiche Tempestive vs. Limitazione delle Risorse di Rete	<p>Invio frequente di notifiche in tempo reale può sovraccaricare le risorse di rete e causare congestione, specialmente durante picchi di utilizzo.</p> <p>Ottimizzare il sistema di notifiche per eventi prioritari (es. conferme prenotazioni) e batch per notifiche secondarie, riducendo il carico di rete senza compromettere l'efficacia delle comunicazioni.</p>

Moderazione Rapida vs. Interventi Complessi	<p>La gestione delle segnalazioni e moderazione dei contenuti richiede interventi rapidi, ma anche una revisione accurata per evitare abusi o errori.</p> <p>Implementare strumenti di moderazione automatizzati per un primo filtro e lasciare a moderatori e admin la verifica dei contenuti più critici, bilanciando velocità e accuratezza.</p>
Chat Accessibile vs. Sicurezza nelle Comunicazioni	<p>La chat tra utenti richiede accessibilità e velocità, ma aumentare la sicurezza (con archiviazione e opzioni di segnalazione) può rallentare l'interazione.</p> <p>Adottare una crittografia leggera per le comunicazioni private e offrire opzioni di archiviazione e segnalazione come funzioni opzionali, bilanciando accessibilità e sicurezza.</p>
Efficienza del Chatbot vs. Profondità del Supporto	<p>Un chatbot efficace deve fornire risposte rapide e semplici, ma anche gestire richieste complesse o avanzate, il che può compromettere la semplicità d'uso.</p> <p>Programmare il chatbot per rispondere rapidamente a domande comuni e offrire l'opzione di trasferire richieste complesse all'assistenza umana per una gestione completa.</p>

1.3: Definizioni, acronimi e abbreviazioni

- **AD:** Activity Diagram
- **RAD:** Requirements Analysis Document
- **RF:** Requisito Funzionale
- **RNF:** Requisito Non Funzionale
- **UI:** User Interface
- **SB:** Social Bridge

1.4: Riferimenti

1. Object-Oriented Software Engineering (Using UML, Patterns, and Java) - Bernd Bruegge & Allen H.
2. Documento di analisi dei requisiti (C18_RAD).

1.5: Overview

Il documento SDD di SocialBridge è suddiviso nelle seguenti sezioni:

- **Introduzione:** Presenta gli obiettivi, i design goal e i trade-off principali.
- **Architettura del Sistema Attuale:** Descrive l'architettura del sistema esistente e le sue limitazioni.
- **Architettura del Sistema Proposto:** Include la decomposizione in sottosistemi, il mapping hardware/software, e le specifiche di persistenza.
- **Servizi dei sottosistemi:** Spiega i servizi forniti dai vari sottosistemi e le condizioni limite.
- **Glossario:** Termini tecnici e definizioni utilizzate nel documento.

2. Architettura del Sistema Attuale

Al momento, non esiste un sistema che integri pienamente le funzionalità offerte da SocialBridge. Alcune piattaforme, come Eventbrite e Friendship Circle, offrono eventi o connessioni sociali, ma nessuna di esse è progettata per rispondere specificamente alle esigenze di inclusione e accessibilità.

3. Architettura del Sistema Proposto

Questo paragrafo contiene la descrizione del sistema SocialBridge sotto il punto di vista architetturale e dei servizi che offre.

3.1: Sintesi del Sistema

SocialBridge è una piattaforma innovativa progettata per promuovere l'inclusività e facilitare la creazione di eventi che uniscono persone con interessi comuni.

Il progetto sfrutta un'architettura moderna basata su MongoDB per la gestione dei dati, Node.js e JavaScript per la logica di back-end e React.js e JavaScript per il front-end.

Grazie a questo stack tecnologico, SocialBridge garantisce una user experience reattiva e scalabile, offrendo agli utenti la possibilità di partecipare o creare eventi accessibili a tutti, favorendo l'inclusione sociale e un ambiente positivo per la connessione tra gli individui.

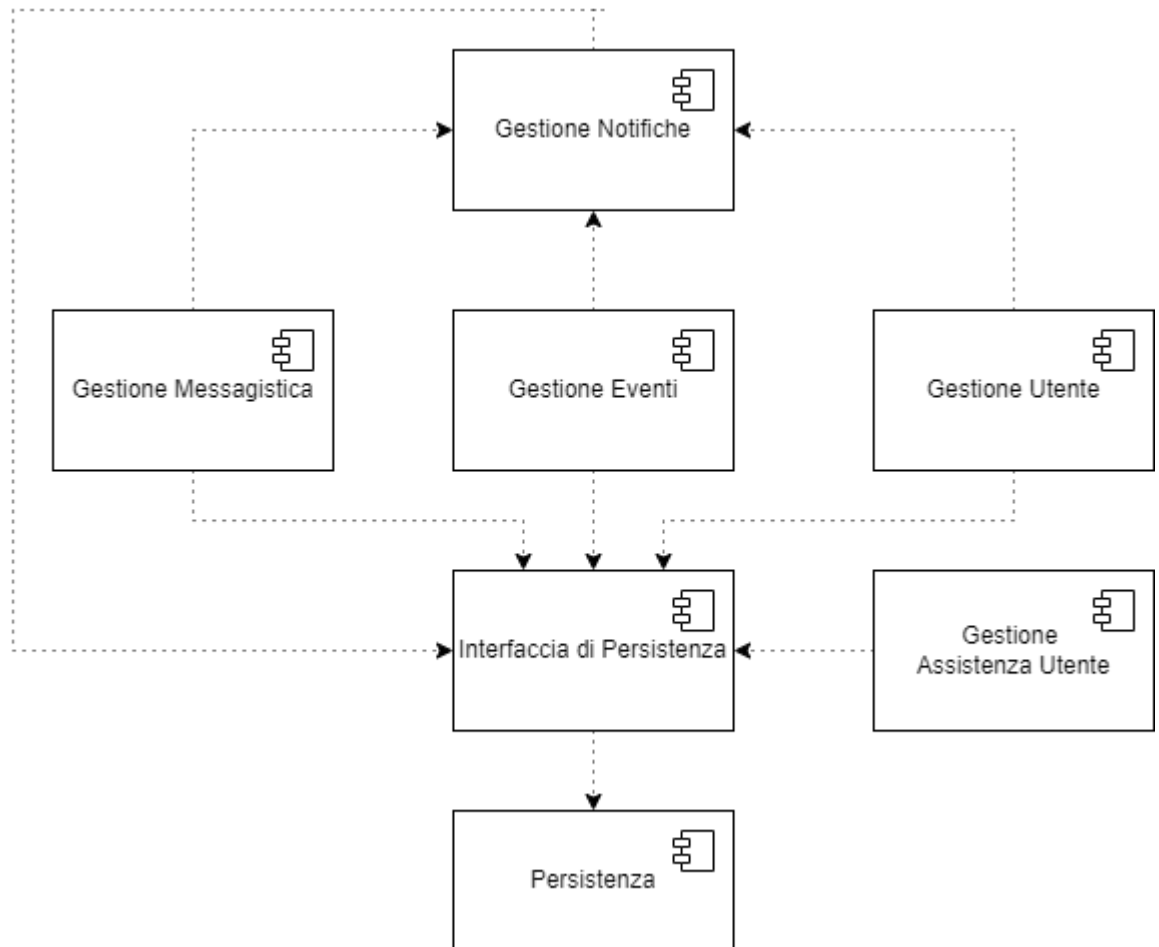
3.2: Decomposizione in Sottosistemi

In questa sezione vengono descritti i vari sottosistemi individuati per il sistema SocialBridge, organizzati in modo da supportare le funzionalità principali richieste dagli utenti. Ogni sottosistema rappresenta un'area di funzionalità specifica, facilitando la gestione e la modularità del sistema.

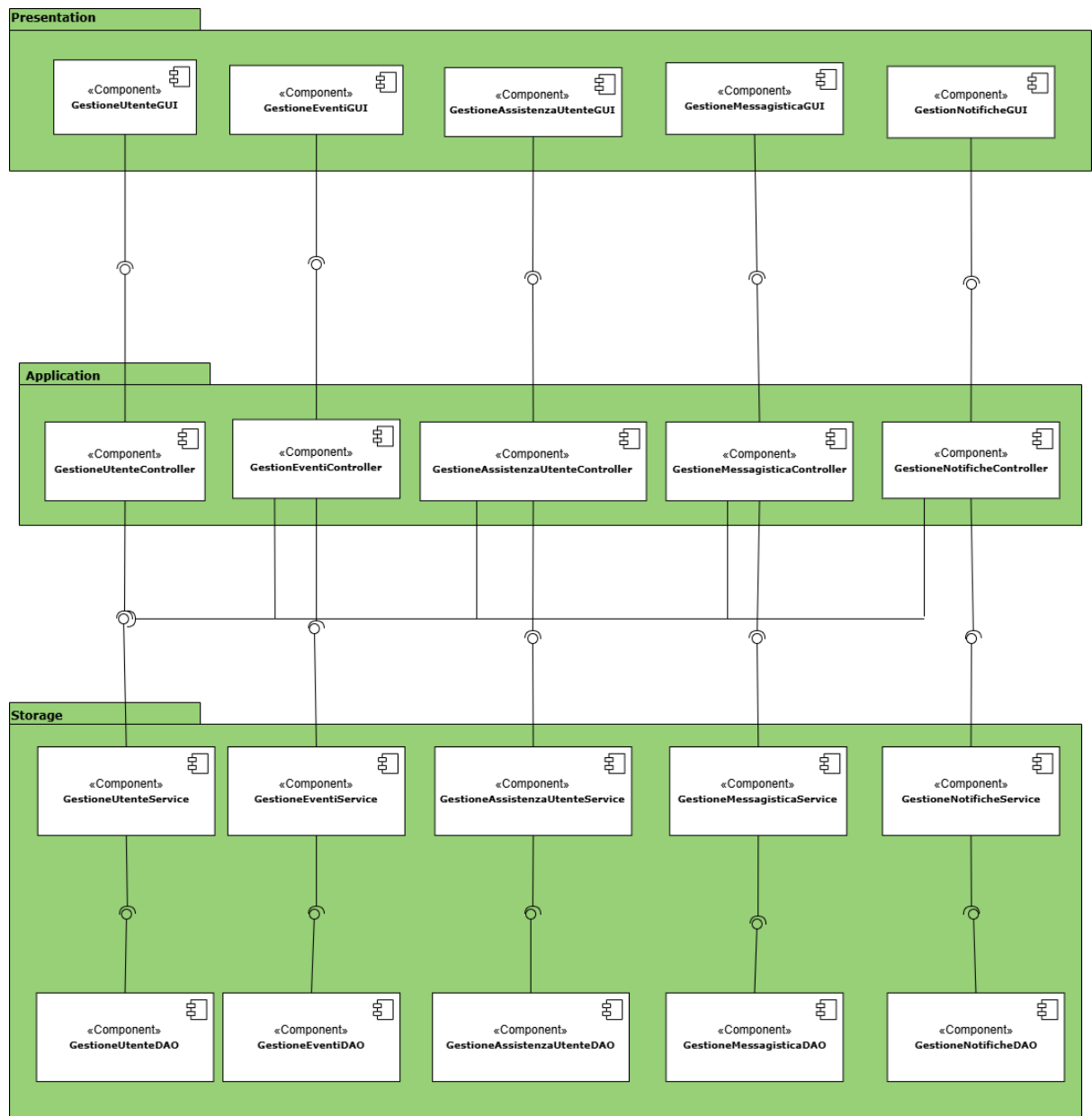
- **Gestione** **Utente:**
Questo sottosistema si occupa di tutte le operazioni di registrazione e autenticazione, inclusa la gestione dei profili utente. Supporta i vari tipi di utenti, come partecipanti, organizzatori e moderatori, gestendo i rispettivi ruoli e autorizzazioni. Inoltre, include funzionalità per l'aggiornamento delle informazioni personali e garantisce la sicurezza dei dati personali degli utenti tramite conferme e feedback immediati per le modifiche effettuate.
- **Gestione** **Eventi:**
Questo sottosistema permette agli organizzatori di creare, modificare e gestire eventi, inclusa la gestione delle iscrizioni e la visualizzazione delle recensioni post-evento per migliorare l'esperienza degli utenti. Per i partecipanti, consente di visualizzare e iscriversi agli eventi, fornendo un sistema di filtraggio per trovare eventi specifici in base alle preferenze di inclusività e accessibilità. Inoltre, questo sottosistema consente di assegnare etichette specifiche agli eventi (es. accessibile, inclusivo) per migliorare la ricerca e la categorizzazione.
- **Gestione** **Messaggistica:**
Questo sottosistema permette la comunicazione tra utenti tramite una chat privata sicura. Gli utenti possono utilizzare la chat per dialogare con altri partecipanti che hanno partecipato agli stessi eventi. Include notifiche per nuovi messaggi e consente agli utenti di segnalare comportamenti inappropriati, migliorando così la sicurezza e l'esperienza di interazione all'interno della piattaforma.
- **Gestione** **Notifiche:**
Il sottosistema delle notifiche gestisce in tempo reale tutte le notifiche per eventi, messaggi e altre attività importanti, come aggiornamenti sugli eventi. Permette agli utenti di ricevere conferme tempestive per azioni rilevanti, come iscrizioni e modifiche dei dati personali, contribuendo a migliorare la comunicazione e l'interattività sulla piattaforma.
- **Gestione** **Assistenza** **Utente:**
Questo sottosistema offre supporto agli utenti tramite un chatbot integrato, progettato per rispondere a domande comuni e fornire assistenza durante la navigazione e l'utilizzo della piattaforma. Il chatbot semplifica l'accesso alle informazioni e guida gli utenti attraverso le diverse funzionalità della piattaforma, offrendo un supporto continuo e accessibile.
- **Interfaccia** **di** **Persistenza:**
L'interfaccia di persistenza funge da strato intermedio tra i vari sottosistemi e il database, facilitando l'accesso ai dati da parte dei vari servizi senza esporre direttamente il database ai singoli sottosistemi. Questo approccio garantisce che i dati siano gestiti in modo coerente e sicuro.
- **Persistenza:**
Questo sottosistema gestisce la persistenza dei dati utilizzando MongoDB. Organizza e mantiene tutte le informazioni rilevanti per il sistema, comprese le informazioni sugli utenti, gli eventi, i messaggi e le notifiche. La persistenza dei dati è garantita tramite

backup regolari e controlli di consistenza, assicurando che le informazioni salvate siano sempre aggiornate e accessibili.

3.2.1: Diagramma delle componenti



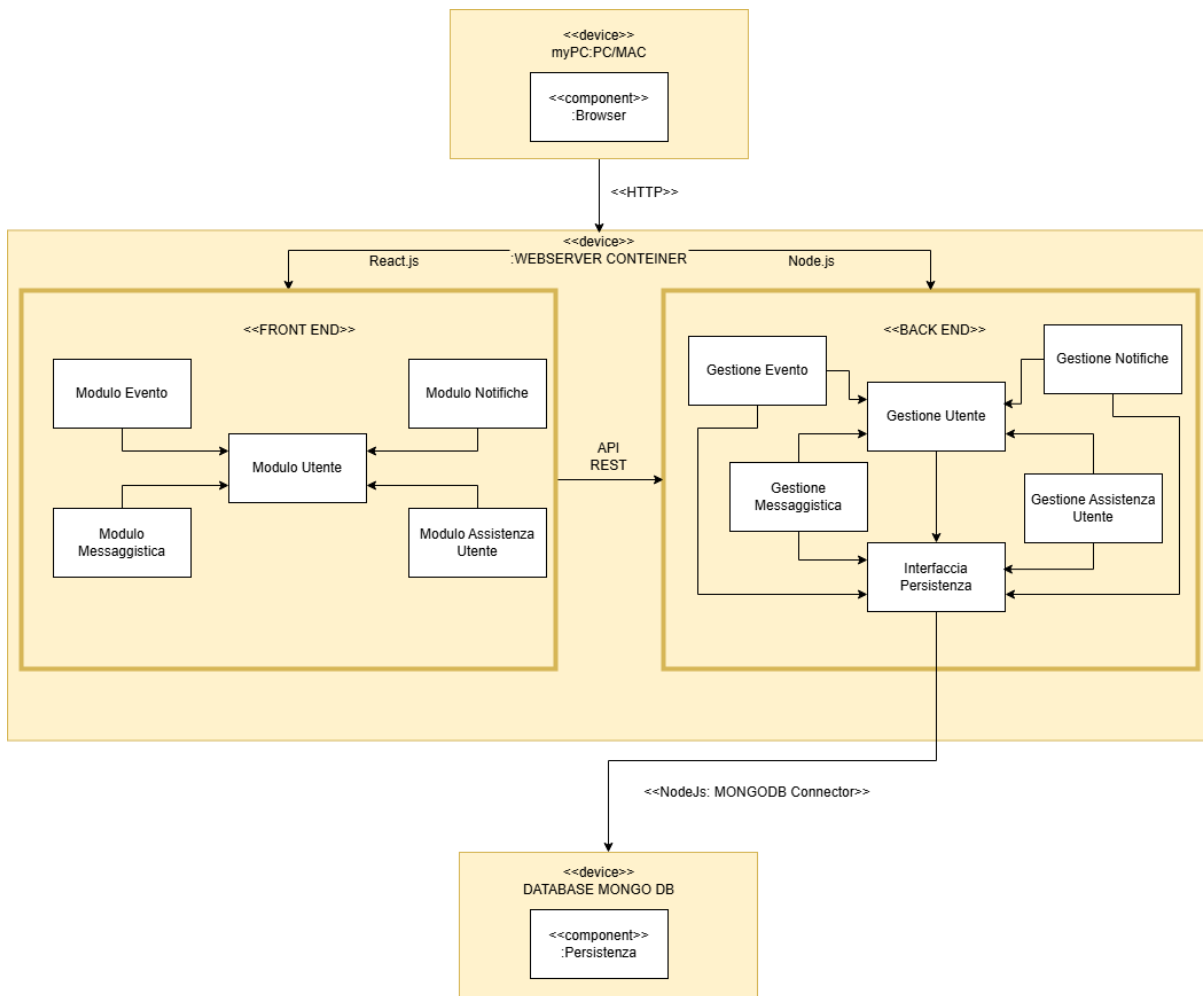
3.2.2: Diagramma architetturale dei sottosistemi



In questo diagramma, invece, vengono mostrati nello specifico le componenti di ogni sottosistema:

- **GUI**, contiene le varie interfacce grafiche che verranno create e mostrate al cliente;
- **Controller**, si occupa della logica per il controllo del sistema;
- **Service**, si occupa della logica di business del sistema;
- **DAO**, si occupa di fornire accesso ai dati persistenti.

3.3: Mapping Hardware e Software



3.4: Gestione dei dati persistenti

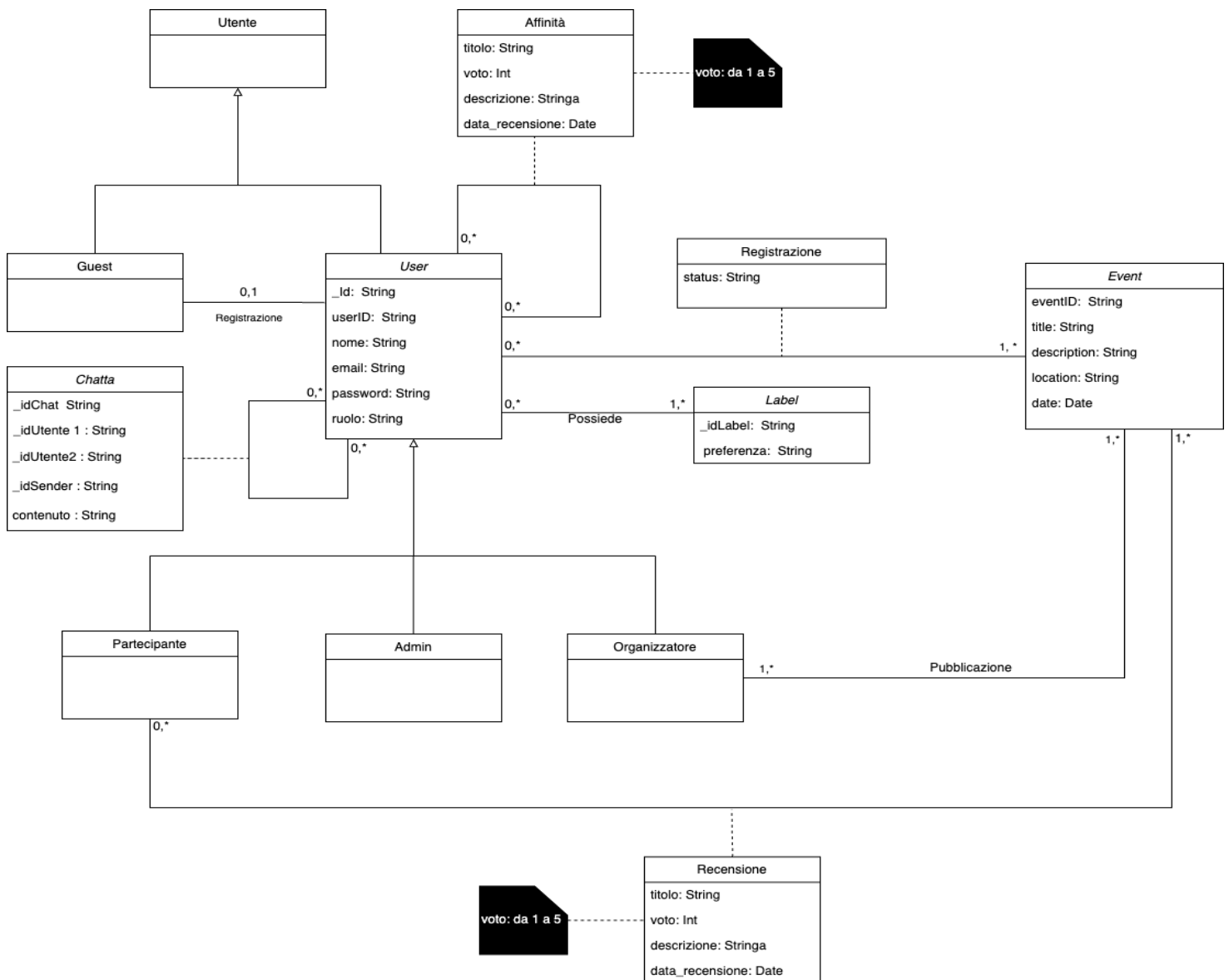
Per la gestione, il salvataggio e le modifiche dei dati persistenti di SocialBridge, si è scelto di utilizzare un database NoSQL, come MongoDB, che offre un'elevata flessibilità nella modellazione dei dati. MongoDB consente una gestione più dinamica delle strutture di dati, risultando adatto per applicazioni come SocialBridge, dove i dati degli utenti e degli eventi possono variare in modo significativo.

Grazie a MongoDB, si riesce a garantire:

- **Consistenza:** attraverso l'uso di transazioni e vincoli a livello di documento che mantengono la coerenza tra i dati correlati, anche durante operazioni complesse.
- **Atomicità:** ogni operazione è garantita come atomica a livello di singolo documento, assicurando che, in caso di errori, nessuna modifica parziale influisca sui dati salvati.
- **Affidabilità:** le funzionalità di backup di MongoDB consentono di preservare copie di sicurezza dei dati, rendendo possibile il ripristino completo in caso di necessità e garantendo la disponibilità continua dei dati per la piattaforma SocialBridge.

MongoDB risulta quindi una scelta ideale per assicurare la scalabilità e la flessibilità necessarie a supportare una piattaforma sociale orientata alla creazione di eventi inclusivi.

CD: Entity Class Diagram

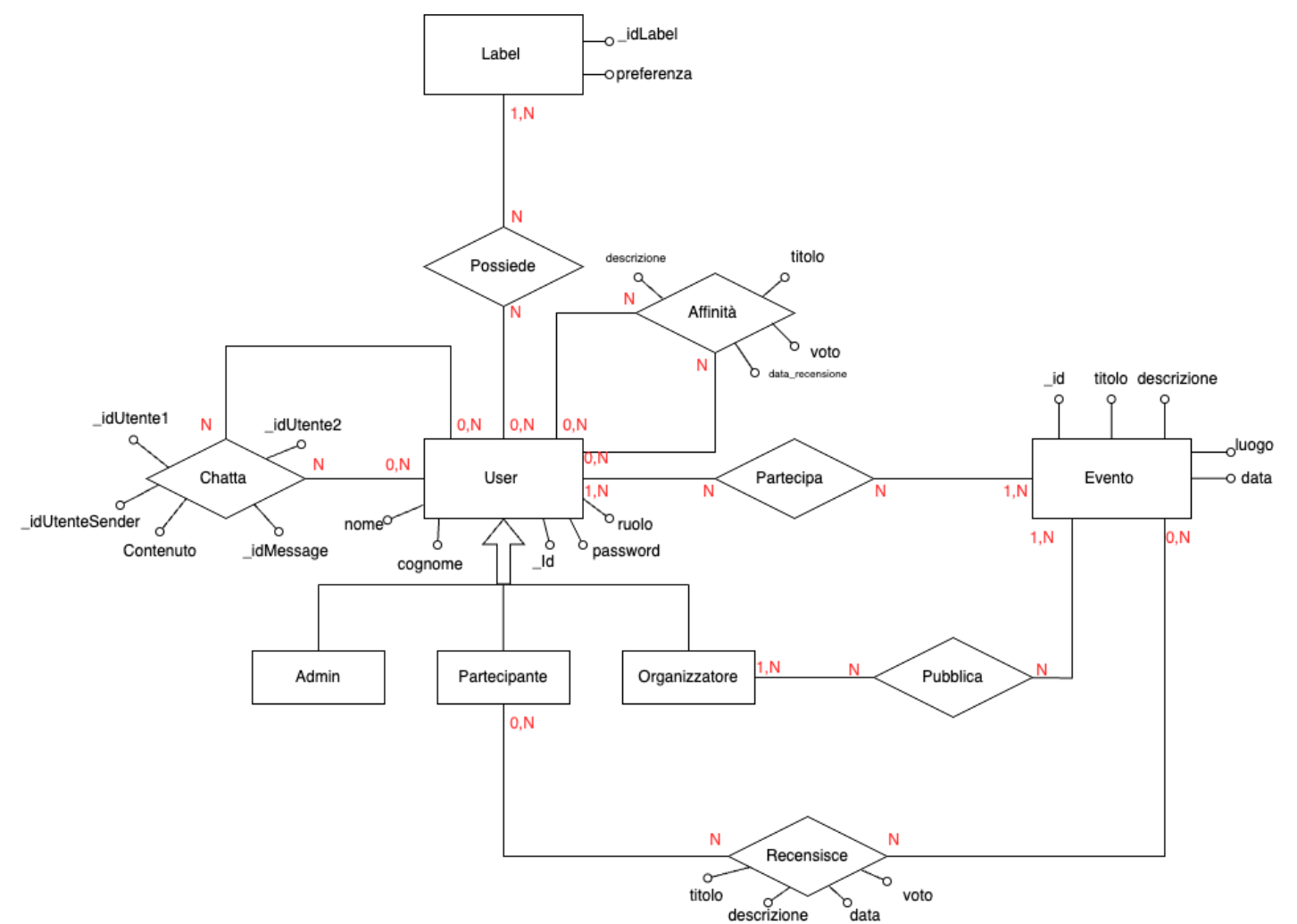


Lista delle entità persistenti

Entità	Descrizione
Utente	Rappresenta un utente generico sulla piattaforma
Guest	Rappresenta un utente non registrato che può visualizzare solo alcune parti pubbliche della piattaforma
User	Un utente registrato con permessi di base, abilitato a partecipare agli eventi e ad interagire con altri utenti
Partecipante	Utente che ha confermato la partecipazione a un evento e che può effettuare una recensione dell’evento
Organizzatore	Utente che può creare e gestire eventi sulla piattaforma

Admin	Utente con permessi amministrativi per la gestione della piattaforma, inclusi controllo e moderazione
Registrazione	Rappresenta il processo e i dettagli di registrazione di un utente alla piattaforma
Evento	Rappresenta un evento creato sulla piattaforma, con informazioni come data, luogo e descrizione
Recensione Evento	Memorizza i feedback degli utenti sugli eventi a cui hanno partecipato, contribuendo a valutare la qualità complessiva di ciascun evento
Affinità Utente	Memorizza i feedback che gli utenti si scambiano al termine di un evento a cui hanno partecipato, fornendo una valutazione dell'affinità e compatibilità tra loro

Modello E-R



Modello dei dati in MongoDB

```
Utente: {
  "_idUtente": "string (obbligatorio, ID univoco dell'utente)",
  "email": "string (obbligatoria)",
  "password": "string (obbligatoria, crittografata)",
  "nome": "string",
  "cognome": "string",
  "ruolo": "string"
}
```

```
Evento: {
  "_idEvento": "string (obbligatorio, ID univoco dell'evento)",
  "titolo": "string (obbligatoria)",
  "descrizione": "string",
  "data": "date",
  "ora": "string",
  "luogo": "string",
  "organizzatore": "string (ID utente)",
  "accessibilita": "string",
  "partecipantiMAX": "integer",
  "pieno": "boolean",
  "labels": "array[string]",
  "partecipanti": "array[string]"
}
```

```
Recensione Evento: {
  "_idRecensione": "string (obbligatorio, ID univoco della recensione)",
  "autoreID": "string (ID utente)",
  "voto": "integer (1-5, obbligatorio)",
  "commento": "string"
}
```

```
Chat: {
  "_idChat": "string (obbligatorio, ID univoco della chat)",
  "mittenteID": "string (obbligatorio, ID univoco della chat)",
  "destinatarioID": "string (obbligatorio, ID univoco della chat)",
  "testo": "string",
  "media": "string",
  "stato": "string",
  "notificato": "boolean",
  "dataInvio": "date"
}
```

```
Registrazione: {
  "_idRegistrazione": "string (ID univoco della registrazione)",
}
```

```
"userID": "string (ID dell'utente registrato)",
"eventoID": "string (ID dell'evento)",
"status": "string (stato della registrazione, es. confermato, in attesa)",
"dataRegistrazione": "date (data della registrazione)"
}
```

```
AffinitaUtente: {
  "_idAffinita": "string (ID univoco dell'affinità,obbligatorio)",
  "user": "string (ID dell'utente)",
  "participantID": "string (ID dell'altro utente)",
  "rating": "integer (1-5, punteggio di affinità)",
  "comment": "string",
  "dataCreazione": "date",

}
```

```
Label: {
  "_idLabel": "string (ID univoco della label, obbligatorio)",
  "nome": "string",
  "descrizione": "string"
}
```

```
Notifica: {
  "_idNotifica": "string (ID univoco della notifica)",
  "utenteId": "string (ID dell'utente destinatario)",
  "messaggio": "string",
  "dataInvio": "date",
  "letto": "boolean"
}
```

Vincoli

- Un evento pubblicato deve avere informazioni minime (titolo, data e luogo).
- Solo gli utenti con il ruolo di "Organizzatore" possono pubblicare eventi.
- L'utente che invia il messaggio deve esistere nel sistema (integrità referenziale tra **User** e **Chat**).
- Un utente deve avere il ruolo di "Partecipante" oppure "Organizzatore" (Escludendo Admin).
- La partecipazione è consentita solo per eventi esistenti nel sistema.
- Un utente può recensire solo eventi a cui ha partecipato.
- Un Organizzatore non può recensire il proprio evento creato.
- Ogni recensione deve includere un voto, un titolo e una data.
- Un utente non può inviare un messaggio a se stesso (vincolo tra mittente e idRicevitore).
- Un utente può iscriversi solo a eventi che sono futuri; quindi, la data dell'evento deve essere successiva alla data corrente al momento dell'iscrizione.

- Un organizzatore non può pubblicare più eventi con lo stesso titolo nello stesso luogo alla stessa data.
- Se un evento viene eliminato, tutte le recensioni e i partecipanti associati devono essere eliminati o deve essere bloccata la cancellazione.

3.5 Sicurezza ed Accessi

Il sistema offre interfacce grafiche diversificate per ogni tipo di utente, in modo che ognuno possa accedere esclusivamente alle funzionalità a lui riservate. Gli utenti guest e quelli registrati possono condividere alcune interfacce, ma l'accesso alle funzionalità complete del sistema è limitato agli utenti che hanno completato la registrazione o che sono autenticati. L'accesso ai dati sensibili e alle funzioni critiche del sistema è permesso solo agli utenti che si autenticano correttamente.

Gli utenti guest hanno accesso parziale al sistema e per sbloccare tutte le funzionalità è necessario registrarsi o autenticarsi. L'autenticazione degli utenti avviene tramite l'inserimento di una email e di una password. Per eseguire operazioni avanzate, come prenotarsi ad un evento, è obbligatoria l'autenticazione.

Il sistema pone una forte enfasi sulla protezione delle informazioni sensibili degli utenti, come password e dati personali. Per garantire la sicurezza di tali informazioni, viene impiegato l'algoritmo di hashing **bcrypt**. **bcrypt** è un algoritmo di hashing progettato per resistere agli attacchi di forza bruta grazie alla sua natura computazionalmente intensiva, ed è particolarmente adatto per proteggere le password degli utenti.

Il sistema si basa su MongoDB, un database NOSQL che offre scalabilità e flessibilità. In MongoDB, le informazioni sensibili, come le password, vengono memorizzate come valori hashati, proteggendo così i dati anche in caso di accesso non autorizzato al database. MongoDB, pur essendo un sistema di database orientato ai documenti, consente di gestire le credenziali e i dati sensibili con un alto livello di sicurezza, integrandosi facilmente con i sistemi di hashing come **bcrypt**.

bcrypt, combinato con una gestione sicura delle chiavi di hash, garantisce che solo gli utenti correttamente autenticati possano accedere ai propri dati. Le password degli utenti vengono hashate prima di essere salvate nel database, e le chiavi di decodifica non sono necessarie poiché l'hashing è un processo unidirezionale, che rende impossibile ricavare la password originale a partire dall'hash.

Inoltre, il sistema si conforma alle normative europee in materia di protezione dei dati personali, in particolare il Regolamento Generale sulla Protezione dei Dati (GDPR) e le leggi sulla privacy elettronica. Ogni utente ha il pieno controllo sui propri dati personali, con il diritto di accesso, modifica, cancellazione e portabilità dei dati. Gli utenti sono informati in modo trasparente su come i loro dati vengono raccolti, utilizzati e protetti, garantendo il rispetto della privacy e la conformità alle normative vigenti.

Solo gli utenti con privilegi amministrativi, come gli Admin, possono interagire con i dati sensibili del sistema, ma esclusivamente dopo aver effettuato una corretta autenticazione. L'autenticazione e la protezione dei dati, supportate da MongoDB e **bcrypt**, assicurano un livello di sicurezza elevato e contribuiscono a mantenere la riservatezza delle informazioni degli utenti.

3.5.1 Matrice degli accessi

Oggetti	Attori	Utente Guest
Gestione Utente		Registrazione
Gestione Eventi		RicercaEvento VisualizzaEvento CondividiEvento
Gestione Assistenza Utente		ContattaChatBot
Gestione Messaggistica		N/A
Gestione Notifiche		N/A

Oggetti	Attori	Utente Registrato Organizzatore	Utente Registrato Partecipante
Gestione Utente		Login Logout VisualizzaAreaUtente ModificaDatiPersonali	Login Logout VisualizzaAreaUtente ModificaDatiPersonali
Gestione Eventi		CreazioneEvento ModificaEvento RicercaEvento VisualizzaEvento CondividiEvento	RicercaEvento VisualizzaEvento CondividiEvento PartecipaEvento
Gestione Assistenza Utente		ContattaChatBot ChiudiChatBot	ContattaChatBot ChiudiChatBot
Gestione Messaggistica		ContattaUtente VisualizzaChat	ContattaUtente VisualizzaChat
Gestione Notifiche		InvioNotificaEvento VisualizzaNotifica	RiceviNotificaEvento VisualizzaNotifica

Attori	Moderatore	Admin
--------	------------	-------

Oggetti		
Gestione Utente	Login Logout VisualizzaAreaUtente ModificaDatiPersonali VisualizzaSegnalazioni	Login Logout VisualizzaAreaUtente ModificaDatiPersonali GestioneUtente VisualizzaSegnalazioni
Gestione Eventi	RimozioneEvento VisualizzaEvento	CreazioneEvento ModificaEvento RimozioneEvento RicercaEvento VisualizzaEvento CondividiEvento AggiuntaUtenteEvento
Gestione Assistenza Utente	SupervisioneChatBot	SupervisioneChatBot MonitoraggioAssistenza
Gestione Messaggistica	N/A	N/A
Gestione Notifiche	InvioNotificaModerazione	InvioNotificaModerazione InvioNotificaNuovoUtente InvioNotificaRimozione InvioNotificaSistema

3.6: Controllo globale del software

Il sistema SocialBridge utilizza un controllo centralizzato basato su eventi (event-driven), dove le azioni vengono attivate da eventi esterni generati dagli utenti (come login, registrazioni, creazione di eventi). Ogni evento è gestito da un listener che lo indirizza al sottosistema appropriato. Dopo aver ricevuto l'evento, il sistema utilizza una logica procedure-driven per completare la richiesta. Le comunicazioni sono sincrone per operazioni critiche e asincrone per notifiche e aggiornamenti. In caso di errori, il sistema gestisce i problemi temporanei autonomamente e notifica quelli persistenti agli amministratori, migliorando l'efficienza e garantendo una risposta rapida agli utenti.

3.7: Condizione Limite

In questo paragrafo verranno rappresentati le boundary conditions relativi all'avvio del sistema, spegnimento del sistema, arresto del sistema e errore di accesso ai dati persistenti.

Avvio del Sistema

Identificativo UCBC_1		UCBC_1 - Avvio del Sistema		Data	12/11/2024
				Vers.	0.1
				Autore	Tutto il team
Descrizione		l'UC permette l'avvio del sistema			
Attore Principale		Admin			
Attori Secondari		N/A			
Entry Condition		Il sistema è spento AND l'admin accende il server AND l'admin invia Il segnale di avvio del sistema			
Exit Condition On success		Il sistema è avviato correttamente			
Exit Condition On failure		Il sistema non è avviato correttamente			
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO					
1	Admin		STEP 1: Si esegue sulla macchina il comando di avvio del sistema		
2	Sistema		STEP 2: il server si collega al database		
3	Sistema		STEP 3: il server si collega ad internet		
4	Sistema		STEP 4: il sistema di mette in attesa per la ricezione di richieste		
Scenario/Flusso di eventi di Errore : Mancata connessione al database					
2.1		Sistema		STEP 1: Invia il segnale di eccezione di mancata connessione al database	
2.2		Admin		STEP 2: Controlla il collegamento al database	
2.3		Admin		STEP 3: Effettua la connessione al database	
Scenario/Flusso di eventi di Errore : Connessione ad internet non riuscita					
3.1		Sistema		STEP 1: Invia il segnale di eccezione di mancata connessione a internet	
3.2		Admin		STEP 2: Controlla la connessione a internet	
3.3		Admin		STEP 3: Effettua la connessione a internet	

Spegnimento del Sistema

Identificativo UCBC_2	UCBC_2 - Spegnimento del Sistema	Data	12/11/2024
		Vers.	0.1
		Autore	Tutto il team
Descrizione	lo UC permette lo spegnimento del sistema		
Attore Principale	Admin		
Attori Secondari	N/A		
Entry Condition	l’admin accede al server AND il sistema è stato precedentemente avviato AND il sistema non è ancora spento		
Exit Condition On success	Il sistema viene spento correttamente		
Exit Condition On failure	Il sistema non viene spento		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Admin	STEP 1: Invia il segnale di spegnimento al sistema	
2	Sistema	STEP 2: Il sistema non viene spento	
Scenario/Flusso di eventi di Errore : ci sono connessione ancora attive (aspetta lo slot di tempo)			
2.1	Sistema	STEP 1: Manda un segnale di eccezione segnalando all’admin che ci sono ancora connessione attive con l’esterno	
2.2	Sistema	STEP 2: Aspetta uno slot di tempo per rispondere ad eventuali richieste dall’esterno senza generare nuove connessione se esse non sono essenziali per la risposta alle già esistenti richieste	
2.3	Sistema	STEP 3: Controlla di nuovo se ci sono connessioni verso l’esterno, se questa condizione è verificata spegne il sistema	
2.4	Sistema	STEP 4: Notifica l’admin dell’avvenuto spegnimento del sistema	
Scenario/Flusso di eventi di Errore : Connessioni ancora attive (interrompe le connessioni)			
2.5	Sistema	STEP 1: Interrompe le connessioni verso l’esterno	
2.6	Sistema	STEP 2: Notifica l’admin dell’avvenuto spegnimento del sistema	

Errore di accessi ai dati persistenti

Identificativo UCBC_3	UCBC_3 - Errore di accessi ai dati persistenti	Data	12/11/2024
		Vers.	0.1
		Autore	Tutto il team
Descrizione	lo UC descrive il comportamento del sistema nel caso vi fosse un errore nell'accesso ai dati persistenti		
Attore Principale	Admin		
Attori Secondari	N/A		
Entry Condition	Si è verificato un errore nell'accesso ai dati persistenti OR i dati sono corrotti		
Exit Condition On success	Il sistema riprende il suo normale funzionamento		
Exit Condition On failure	Il sistema non riprende il suo normale funzionamento		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Sistema	STEP 1: Notifica all'admin che non è stato in grado di accedere ai dati persistenti	
2	Sistema	STEP 2: Smette di processare le richieste	
3	Admin	STEP 3: Include (UCBC_2)	
4	Admin	STEP 4: Risolve l'errore e ripristina l'accesso ai dati	
5	Admin	STEP 5: Include (UCBC_1)	

Identificativo UCBC_4	UCBC_4 - Crash del sistema	Data	12/11/2024
		Vers.	0.1
		Autore	Tutto il team
Descrizione	lo UC descrive il crash del sistema e ne permette la risoluzione		
Attore Principale	Admin		
Attori Secondari	N/A		
Entry Condition	Il sistema è stato precedentemente avviato AND il sistema va in crash		
Exit Condition On success	Vengono avviate procedure di gestione delle eccezioni AND il sistema viene riavviato e continua la sua normale esecuzione		
Exit Condition On failure	Il sistema non continua la sua normale esecuzione		
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO			
1	Sistema	STEP 1: Notifica all’admin di avvenuta eccezione	
2	Admin	STEP 2: Include (UCBC_2)	
4	Admin	STEP 3: Avvia le procedure di gestione degli errori	
5	Admin	STEP 4: Include (UCBC_1)	

Crash del sistema

4. Servizi dei Sottosistemi

Servizi del sottosistema Gestione Utente

SERVIZIO	DESCRIZIONE
Registrazione Utente	Permette la registrazione di un utente
Login	Permette l'accesso alla piattaforma ad un utente precedentemente registrato
Logout	Permette la disconnessione dalla piattaforma ad un utente che precedentemente è acceduto
Modifica Dati Personali	Permette ad un utente di poter modificare i dati personali del proprio account
Visualizzazione Area Utente	Permette all'utente di visualizzare una pagina con tutti i suoi dati.
Visualizza Segnalazioni	Permette all'utente admin o moderatore di visualizzare le segnalazioni relative a un utente.

Servizi del sottosistema Gestione Evento

SERVIZIO	DESCRIZIONE
Creazione Evento	Permette all'utente di creare un evento sulla piattaforma.
Modifica Evento	Permette all'utente di modificare un evento sulla piattaforma.
Visualizzazione Evento	Permette all'utente di visualizzare un evento sulla piattaforma.
Ricerca Evento	Permette all'utente di ricercare un evento, tramite città o tipologia dell'evento, sulla piattaforma.
Partecipa Evento	Permette all'utente di partecipare ad un evento sulla piattaforma.
Condivisione Evento	Permette all'utente di condividere un evento sulla piattaforma.
Rimozione Evento	Permette all'utente organizzatore dell'evento di rimuovere un il suo evento dalla piattaforma.

Aggiunta Utente Evento	Permette all'utente admin di aggiungere un utente che ha richiesto assistenza per difficoltà nell'accesso all'evento.
-------------------------------	---

Servizi del sottosistema Gestione Messaggistica

SERVIZIO	DESCRIZIONE
Contatta Utente	Permette all'utente di scrivere con un altro sulla piattaforma.
Visualizza Chat	Permette all'utente di visualizzare la sua chat con un altro utente.

Servizi del sottosistema Gestione Assistenza Utente

SERVIZIO	DESCRIZIONE
Contatta chatbot	Permette ad un utente che ha fatto accesso alla piattaforma di poter contattare il chatbot ed effettuare richieste di assistenza per lo svolgimento di operazioni.
Chiudi chatbot	Permette ad un utente che ha precedentemente contattato un chatbot di chiudere l'assistenza.
Supervisione ChatBot	Monitora le risposte del chatbot per garantire che siano accurate e utili. Se il chatbot non può rispondere a una richiesta, la inoltra al supporto umano.
Monitoraggio Assistenza	Consente al team di assistenza di controllare le richieste degli utenti e valutare le risposte del chatbot o degli operatori per migliorare l'efficacia del servizio.

Servizi del sottosistema Gestione Notifiche

SERVIZIO	DESCRIZIONE
Invia notifiche evento	Informa i partecipanti sugli aggiornamenti degli eventi cui sono iscritti.
Invia Notifica Nuovo Utente	Invia una notifica agli utenti per informarli dell'aggiunta di nuovi eventi o contenuti rilevanti, includendo dettagli di base come il titolo e la data dell'evento.

Invia Modifica Moderazione	Avvisa gli utenti quando i moderatori apportano modifiche a eventi o contenuti, specificando il motivo della modifica.
Ricevi Notifica Evento	Fornisce aggiornamenti e promemoria ai partecipanti sugli eventi cui sono iscritti, come modifiche di programma o conferme di iscrizione.
Invia Notifica Rimozione	Informa gli utenti della rimozione di un evento o contenuto, specificando il motivo e fornendo istruzioni su eventuali azioni da intraprendere.
Visualizza Notifiche	Consente agli utenti di accedere a una sezione dove visualizzare tutte le notifiche ricevute, con la possibilità di filtrare per tipo o data.
Invia Notifica Sistema	Invia notifiche generali agli utenti su aggiornamenti della piattaforma, manutenzioni programmate o modifiche alle politiche d'uso.

5. Glossario

Questa sezione raccoglie i termini chiave utilizzati nel documento, accompagnati da definizioni utili per comprendere concetti specifici. Fornisce una guida rapida e chiara al linguaggio tecnico, offrendo una panoramica esaustiva e accessibile. Il glossario ha l'obiettivo di garantire chiarezza e migliorare la comprensione dei termini fondamentali presenti nel SDD.

- **MongoDB**: Database NoSQL utilizzato per la gestione dei dati.
- **bcrypt**: Algoritmo di hashing usato per la protezione delle password.
- **RNF**: Requisito non funzionale.
- **GUI**: Interfaccia grafica utente