



UNIVERSITÀ DEGLI STUDI DI PALERMO
DIPARTIMENTO DI INGEGNERIA
CORSO DI LAUREA TRIENNALE IN INGEGNERIA INFORMATICA

Paper Review

System Design Document

Studenti

Leonardo Giovanni Caiezza
Diego Corona
Luca Gaetani
Daniele Orazio Susino

Docente

Prof.ssa Valeria Seidita

Anno accademico

2024 - 2025

Indice

1	Obiettivo del sistema	2
2	Architettura del software attuale	2
3	Obiettivi di progettazione	2
4	Architettura Software Proposta	3
4.1	Panoramica	3
4.2	Requisiti minimi per l'utilizzo del software proposto	3
4.3	Scomposizione in sottosistemi	3
4.3.1	Gestione Notifiche	4
4.3.2	Gestione Account	4
4.3.3	Gestione Conferenza	4
4.3.4	Gestione Revisioni	4
4.3.5	Presentazione Articolo	4
4.3.6	Gestione Paper Definitivi	4
4.4	Mappatura degli oggetti nei sottosistemi	5
4.4.1	Gestione Notifiche	5
4.4.2	Gestione Account	6
4.4.3	Gestione Conferenza	7
4.4.4	Gestione Revisioni	8
4.4.5	Presentazione Articolo	9
4.4.6	Gestione Paper Definitivi	10
4.5	Mappatura Hardware/Software	11
5	Gestione dei Dati Persistenti	12
5.1	Schema E-R	12
5.2	Modello Relazionale	13
5.3	Struttura delle tabelle	14
5.3.1	Utente	14
5.3.2	Notifica	14
5.3.3	Topic	14
5.3.4	Conferenza	15
5.3.5	Invito	16
5.3.6	Paper	16
5.3.7	Revisione	17
5.3.8	Proceeding	17
5.3.9	Ruolo_conferenza	18
5.3.10	TopicUtente	18
5.3.11	CoAutoriPaper	18
6	Copyright e diritto d'autore	19

1 Obiettivo del sistema

L'obiettivo del sistema è **supportare** e automatizzare l'intero **processo organizzativo di una conferenza** scientifica o accademica, facilitando l'interazione tra le diverse parti coinvolte.

Il sistema consente agli autori di sottoporre i propri papers, ai revisori e sottorevisori di valutare i lavori assegnati, ai chair di creare e gestire le conferenze nonché supervisionarne il processo decisionale, e agli editori di curare la pubblicazione finale dei papers.

Un aspetto innovativo del sistema è l'integrazione di un **modello linguistico di grandi dimensioni** (LLM) per la gestione automatica dei tag semantici, che aiuta a classificare i paper e ad assegnarli ai revisori più adatti. In questo modo, il sistema migliora l'efficienza, la coerenza e la qualità del processo di revisione e pubblicazione.

2 Architettura del software attuale

Si suppone che al momento l'**azienda non adotti alcun software** dotato delle funzionalità offerte dal Sistema.

Le **attività** che quest'ultimo mira ad automatizzare **venivano** infatti **gestite manualmente** dalle diverse figure coinvolte nei processi, con conseguente maggiore dispendio di tempo e risorse.

3 Obiettivi di progettazione

- Il **sistema e il DBMS devono essere sempre attivi**, per permettere agli utenti di accedere alle informazioni in qualunque momento. Tuttavia, al di fuori dell'orario di lavoro, è tollerabile un periodo di inattività di 5 minuti al giorno per permettere piccoli interventi di manutenzione.
- Le **password** degli account devono essere **criptate** all'interno del database.
- Il software **non deve permettere** l'inserimento di **input sintatticamente scorretti** da parte degli utenti.
- La **piattaforma deve essere intuitiva**, facile da usare e immediata, in modo da risultare accessibile a un ampio e variegato gruppo di utenti.
- Il sistema deve poter **fronteggiare una perdita di connessione** segnalando questa all'utente, assicurando la coerenza e la persistenza dei dati immagazzinati.
- In caso di errore, il sistema deve permettere all'utente di **riprendere il lavoro** dal punto in cui si era interrotto.
- Il sistema deve essere organizzato in modo tale da **minimizzare** il numero di **comunicazioni con il DBMS**.
- Il sistema deve poter rispondere alle richieste dell'utente al più entro tre secondi.
- Il sistema ed il DBMS dovranno avere un **uptime** quanto più vicino possibile al 100% con un impegno **best effort**, per permettere una fruizione continua dei servizi e delle informazioni per gli utenti.
- Il sistema deve essere **protetto da vulnerabilità note**, come SQL injection. Ciò comporta l'adozione di pratiche di sviluppo sicuro, come la validazione e sanificazione dell'input e l'uso di query parametrizzate.

4 Architettura Software Proposta

4.1 Panoramica

Per l'implementazione del sistema è stata scelta un'architettura ibrida Three Tier-Repository, in quanto garantisce scalabilità, manutenibilità, modularità e ottime performance. Essa è composta da:

- **Interface Layer:** si occupa dell'interfaccia utente.
- **Logic Tier:** contiene le regole di business e gestisce i flussi dell'applicazione.
- **Data Tier con pattern Repository:** gestisce la persistenza e l'accesso ai dati. Nel pattern Repository, si crea un'astrazione per accedere a una sorgente di dati, nascondendo la complessità dell'accesso a database o API esterne.

La combinazione dei due pattern architetturali consente la **separazione delle responsabilità**, testabilità e flessibilità del software.

L'interfaccia utente, i controller e l'interfaccia al database e alle API di ciascun sottosistema risiedono sullo stesso nodo, mentre il sottosistema del database e delle API è ospitato su un nodo separato.

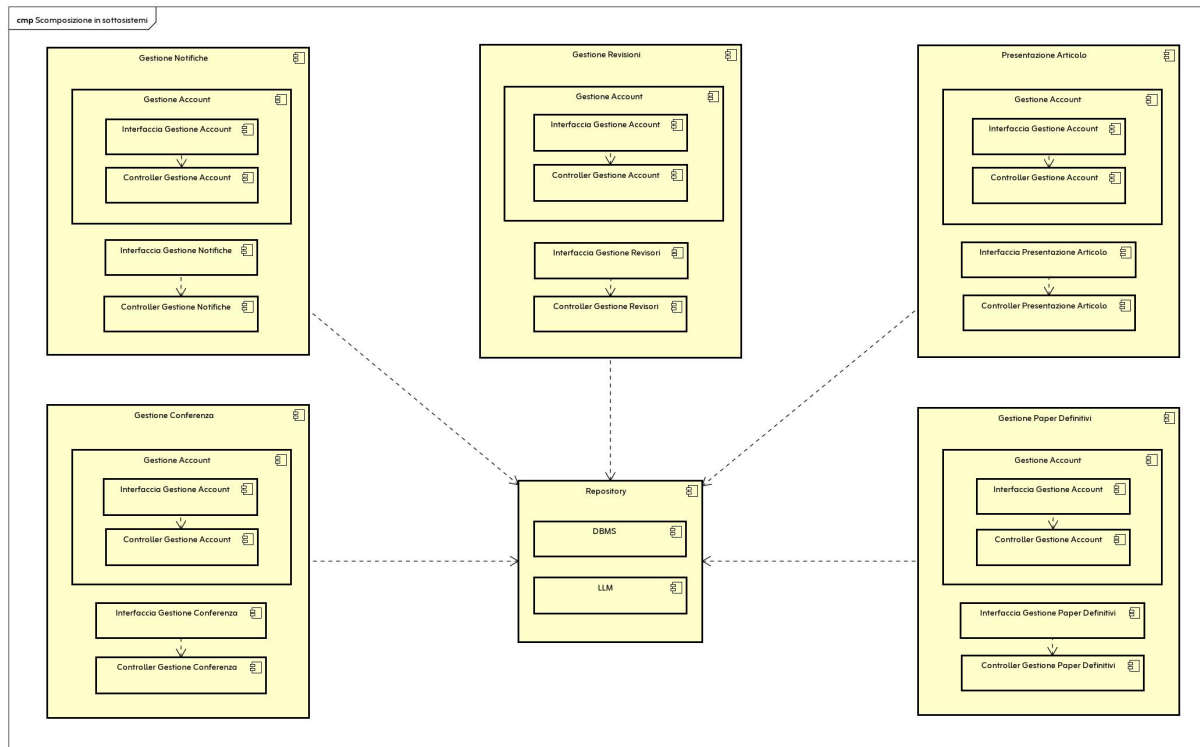
4.2 Requisiti minimi per l'utilizzo del software proposto

Per funzionare correttamente, il sistema richiede una **connessione a Internet stabile** per consentire la comunicazione tra i sottosistemi, il DBMS e le API, nonché per inviare le email per il recupero delle credenziali. Inoltre, è necessaria una **casella di posta elettronica** per ogni utente che vuole utilizzare il sistema.

Inoltre, per il **nodo server** è indispensabile l'impiego di una **scheda video NVIDIA con CUDA cores**, al fine di gestire l'elaborazione dei dati della LLM e garantire prestazioni ottimali.

4.3 Scomposizione in sottosistemi

Di seguito viene rappresentata la suddivisione in sottosistemi del sistema proposto:



4.3.1 Gestione Notifiche

Il sottosistema Gestione Notifiche comprende i moduli che permettono l'**automatizzazione delle notifiche** e la **ricezione degli inviti** da parte degli utenti.

In particolare, contiene i sottosistemi dedicati alla **gestione delle scadenze imminenti** e delle operazioni da eseguire non appena si rendano disponibili nuove attività relative ai paper.

Inoltre, contiene i sottosistemi che permettono all'utente di partecipare ad una conferenza mediante un codice invito, consentendo loro di accettare o rifiutare gli inviti.

Infine, si occupa della gestione dei sottosistemi per l'**archiviazione e la visualizzazione delle notifiche** e degli inviti da parte degli utenti.

4.3.2 Gestione Account

Il sottosistema Gestione Account contiene i sottosistemi che permettono agli utenti di effettuare il login, registrarsi, **autenticarsi**, recuperare le credenziali, effettuare il logout e modificare la password del proprio profilo.

4.3.3 Gestione Conferenza

Il sottosistema Gestione Conferenza contiene i sottosistemi che permettono ai chair di **creare le conferenze e gestirne le relative scadenze**, i parametri, il rate di approvazione e la scala del punteggio dei paper.

Oltre a questo, contiene i sottosistemi che consentono ai chair di **invitare** altri chair, revisori e l'editore per una specifica conferenza.

Infine, contiene i sottosistemi che permettono ai chair di **assegnare manualmente o in modo automatico i paper** ai revisori, e di segnalare eventuali revisori in conflitto con determinati paper.

4.3.4 Gestione Revisioni

Il sottosistema Gestione Revisioni contiene i sottosistemi che permettono ai revisori ed ai sotto-revisori di **presentare e modificare revisioni** fino alla relativa scadenza e di segnalare un possibile plagio.

Inoltre, contiene i sottosistemi che consentono ai revisori di **visualizzare le revisioni dei loro colleghi** (esclusivamente dopo aver pubblicato la propria revisione), nonché i sottosistemi che permettono di **invitare sotto-revisori** per un paper a loro assegnato.

4.3.5 Presentazione Articolo

Il sottosistema Presentazione Articolo contiene i sottosistemi che permettono agli autori di **presentare e modificare la propria sottomissione** entro le relative scadenze, ed eventualmente di rinunciare alla presentazione.

Inoltre, contiene i sottosistemi che permettono la **selezione automatica (tramite LLM) e manuale delle parole chiave** dei paper.

4.3.6 Gestione Paper Definitivi

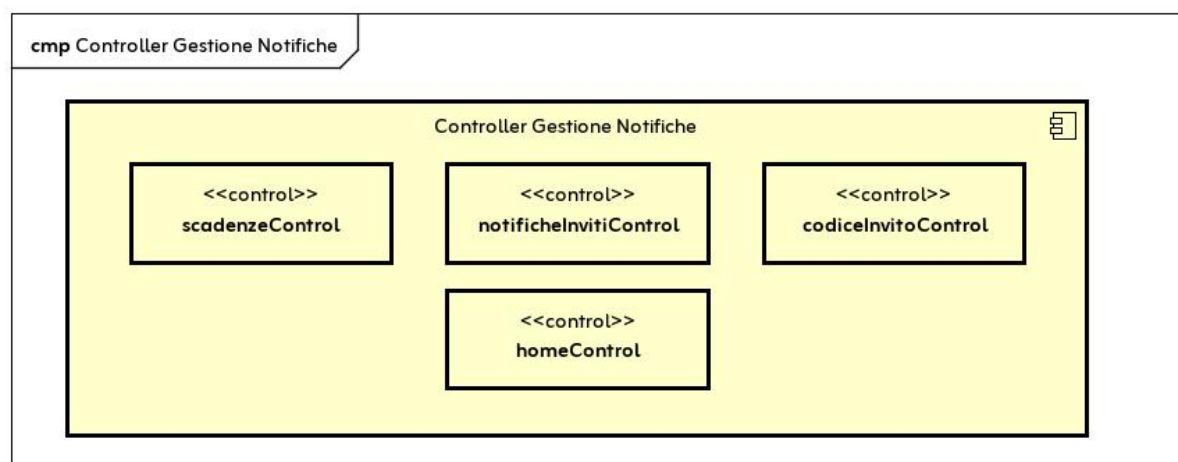
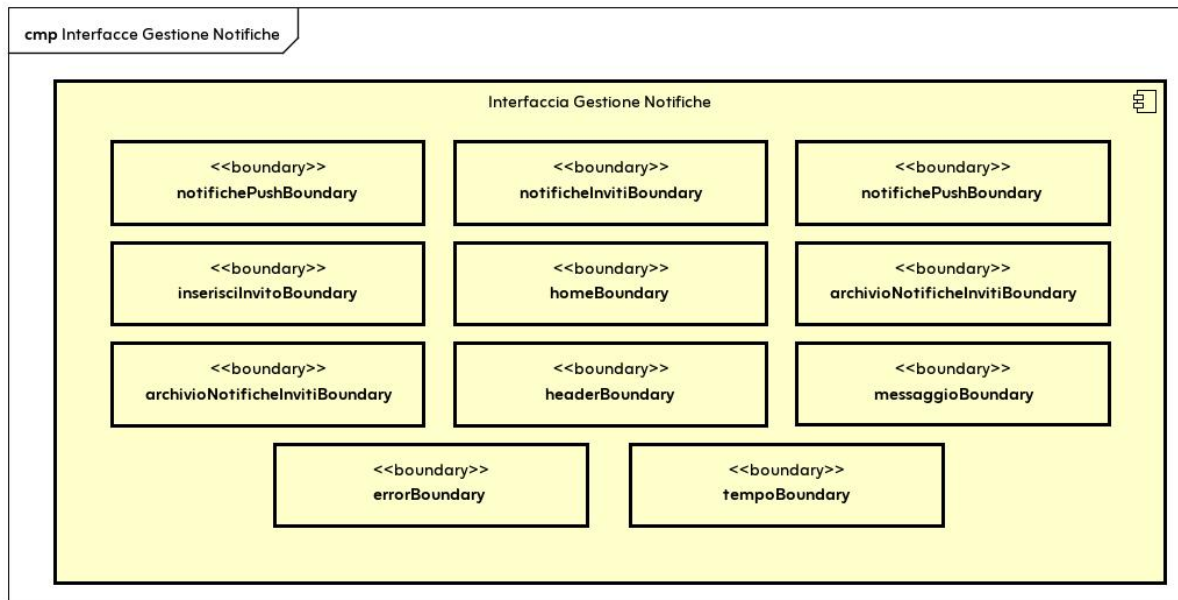
Il sottosistema Gestione Paper Definitivi contiene i sottosistemi che permettono ad un **editore di scaricare i paper**, richiedere delle correzioni agli autori e pubblicare i **proceedings** (versione definitiva).

Inoltre, contiene i sottosistemi che permettono ad un autore di **caricare la versione corretta** del proprio paper.

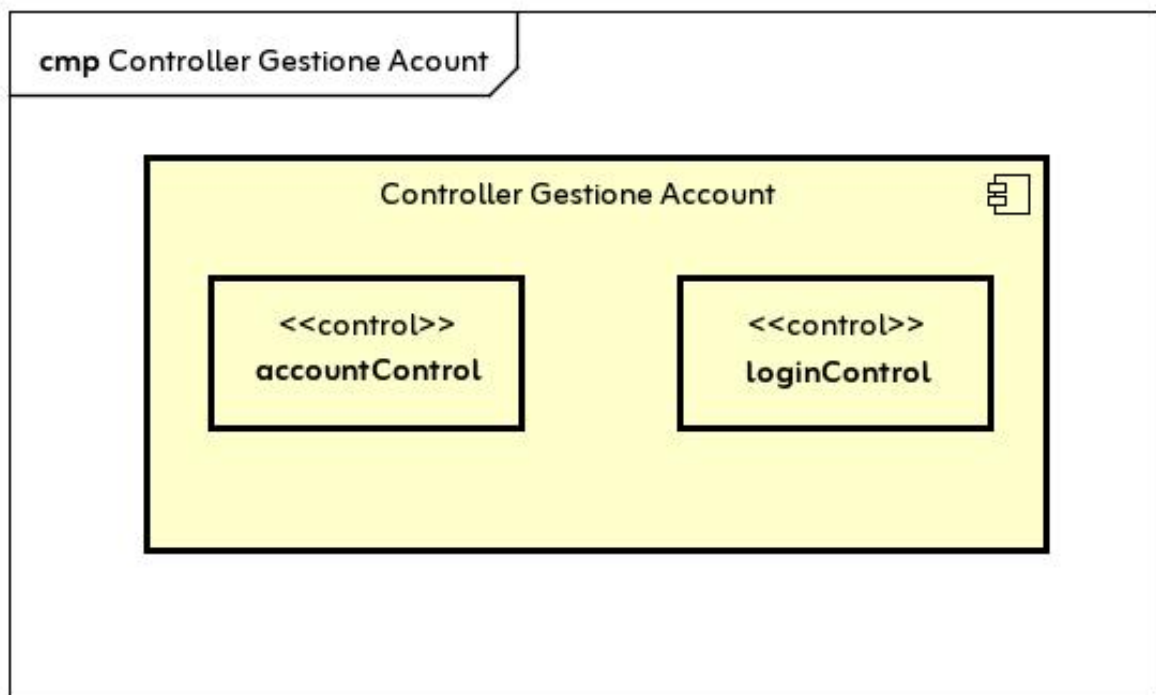
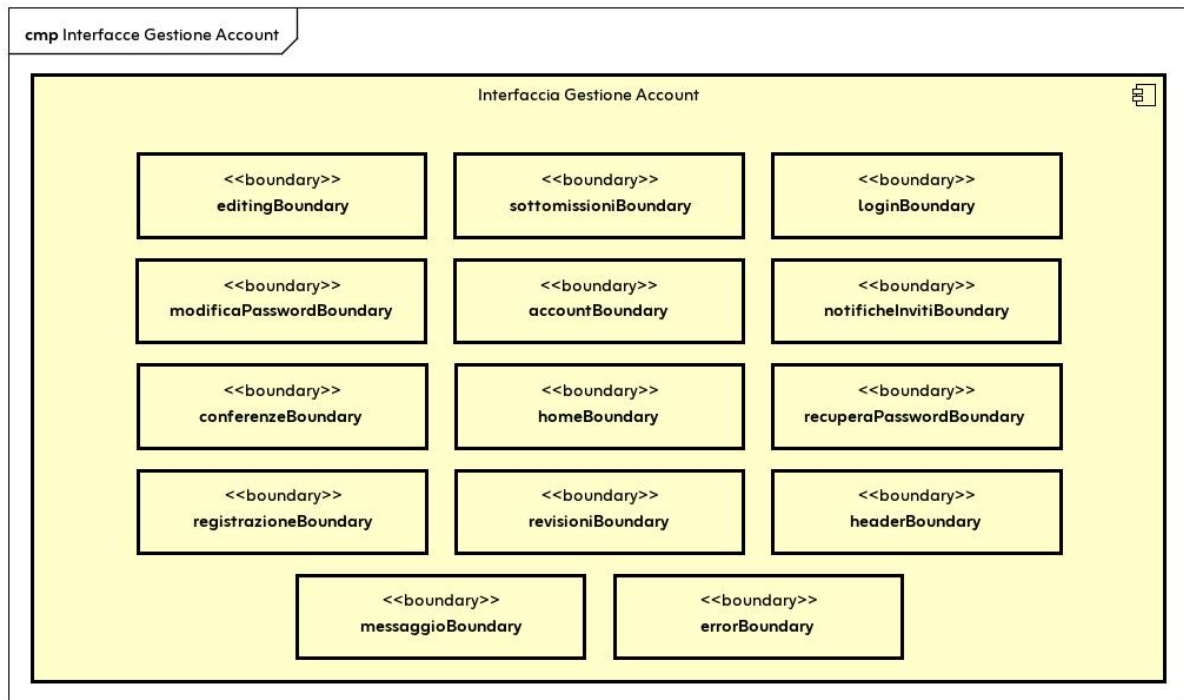
4.4 Mappatura degli oggetti nei sottosistemi

Di seguito sono riportate le funzionalità offerte dai sottosistemi:

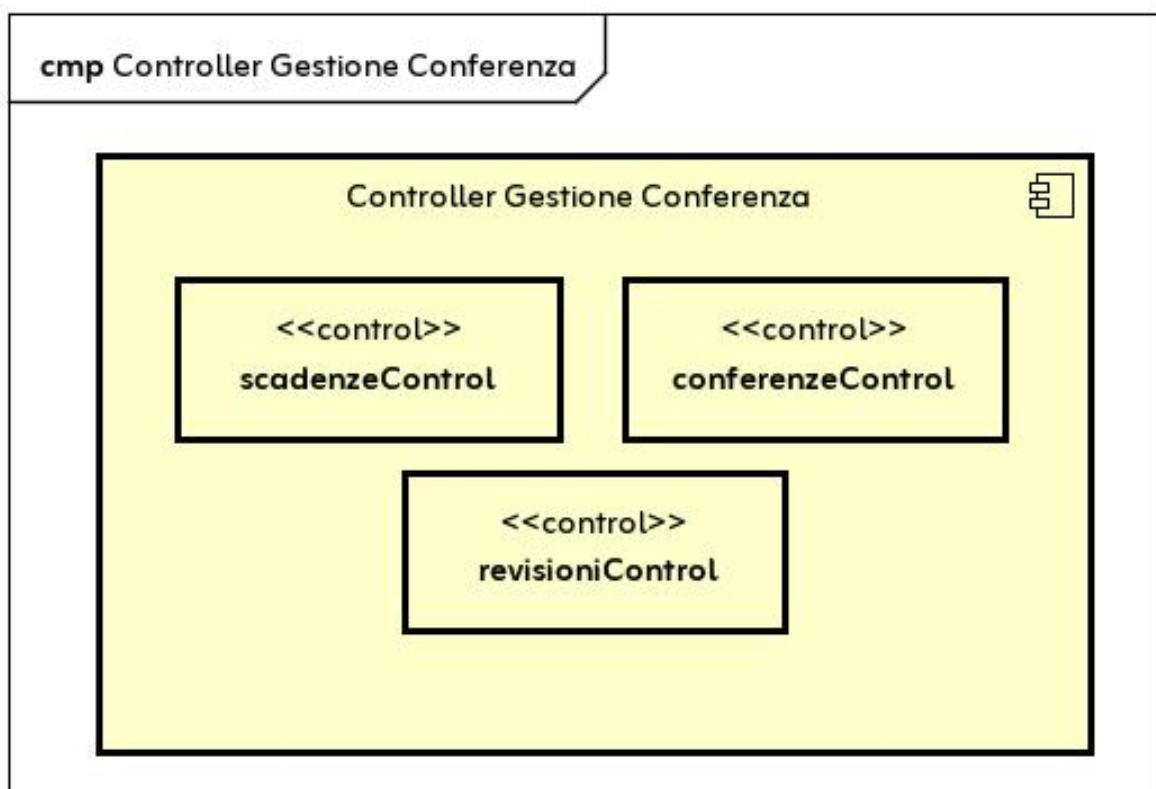
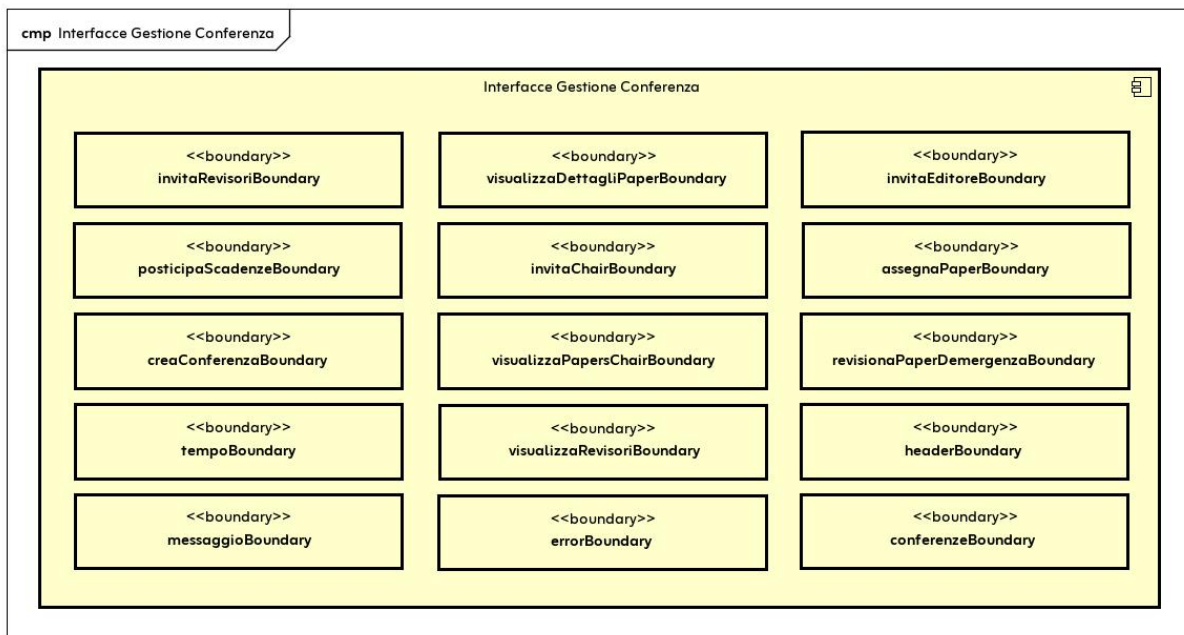
4.4.1 Gestione Notifiche



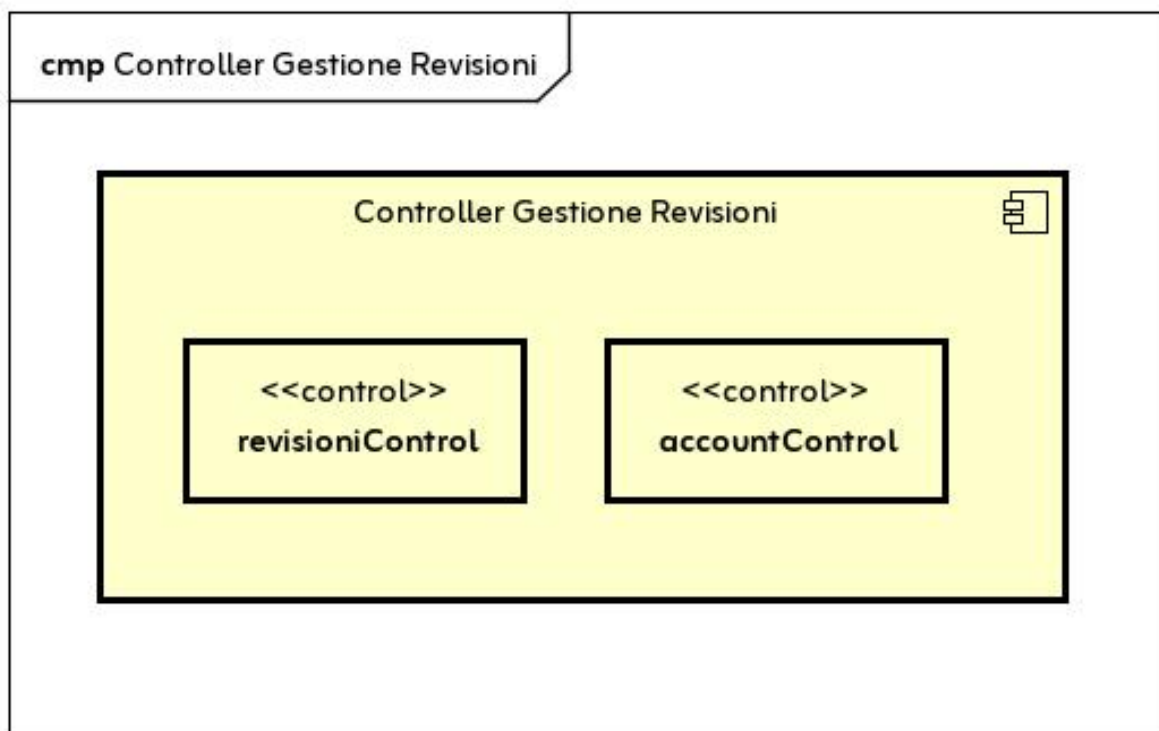
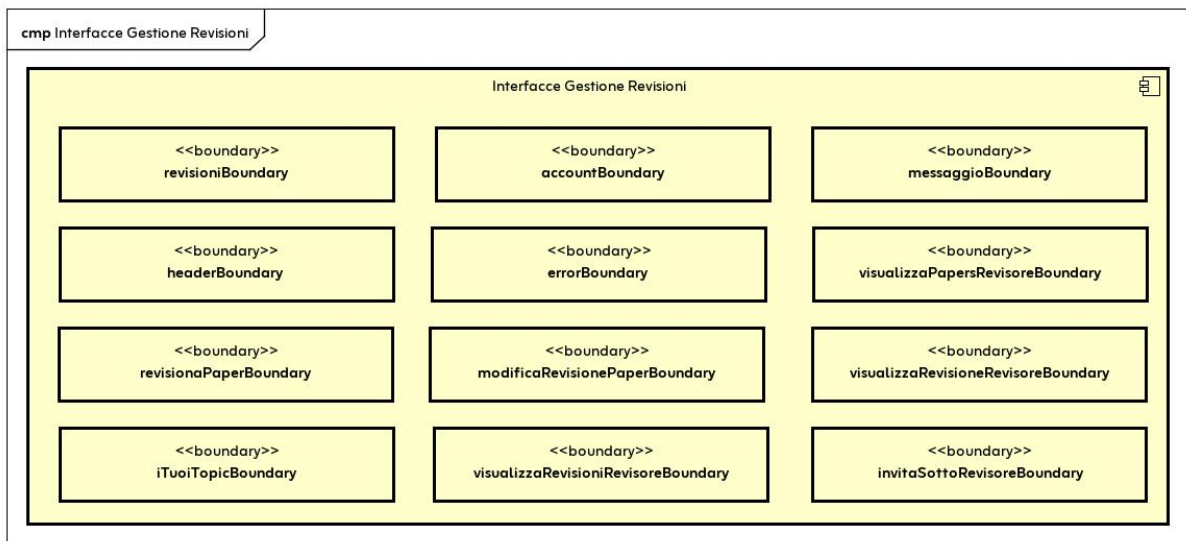
4.4.2 Gestione Account



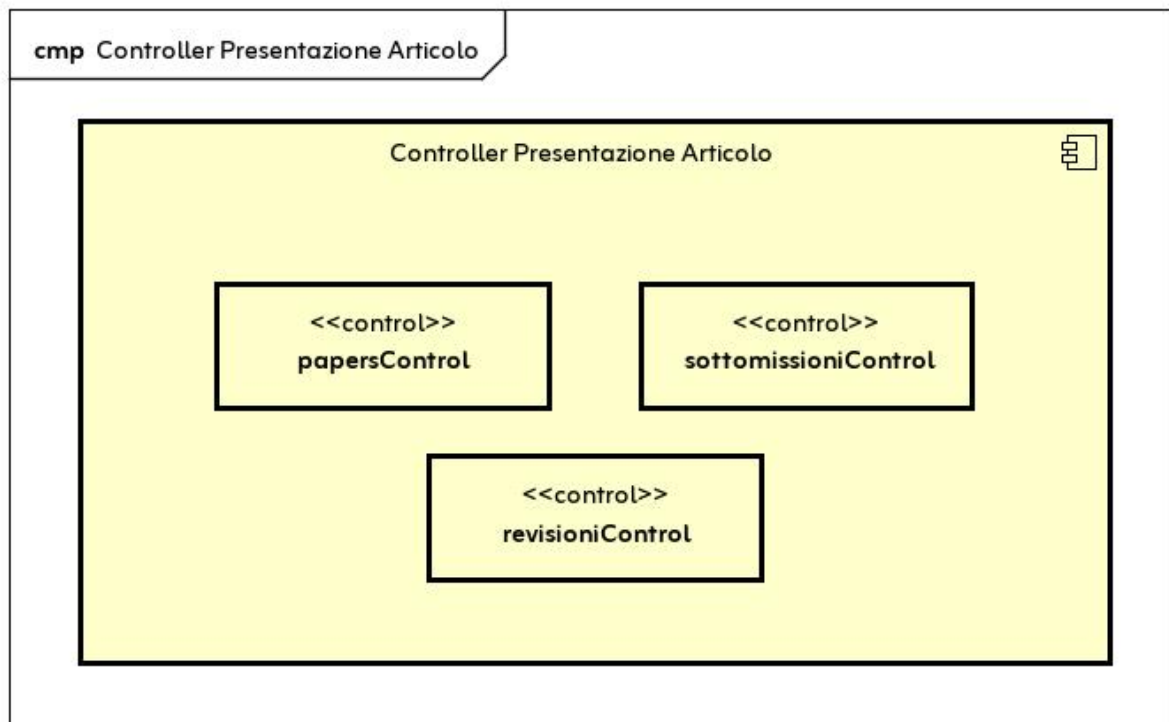
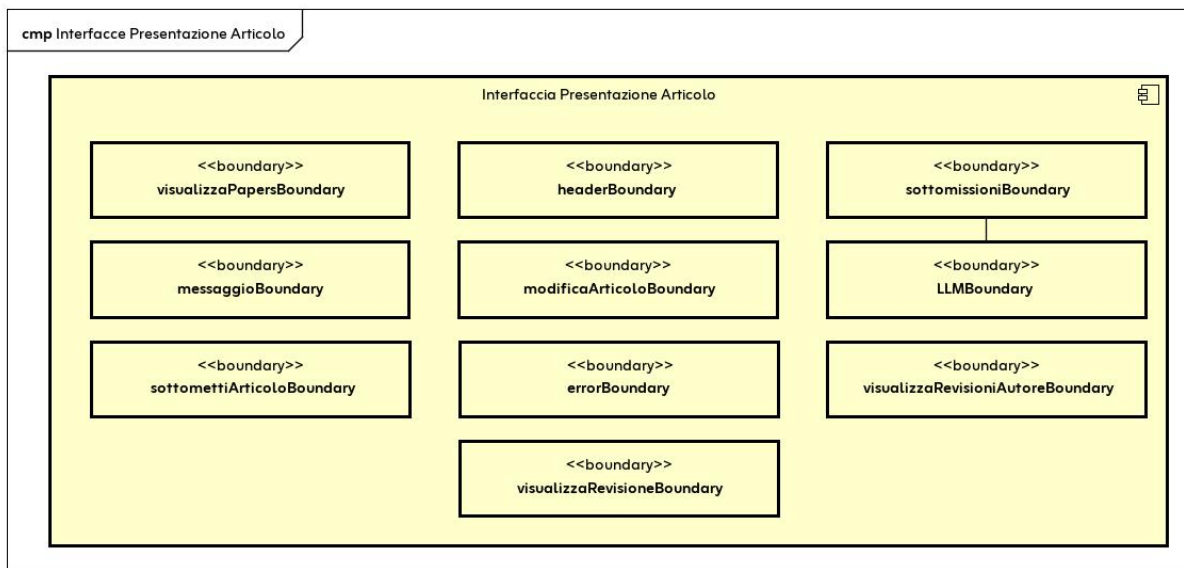
4.4.3 Gestione Conferenza



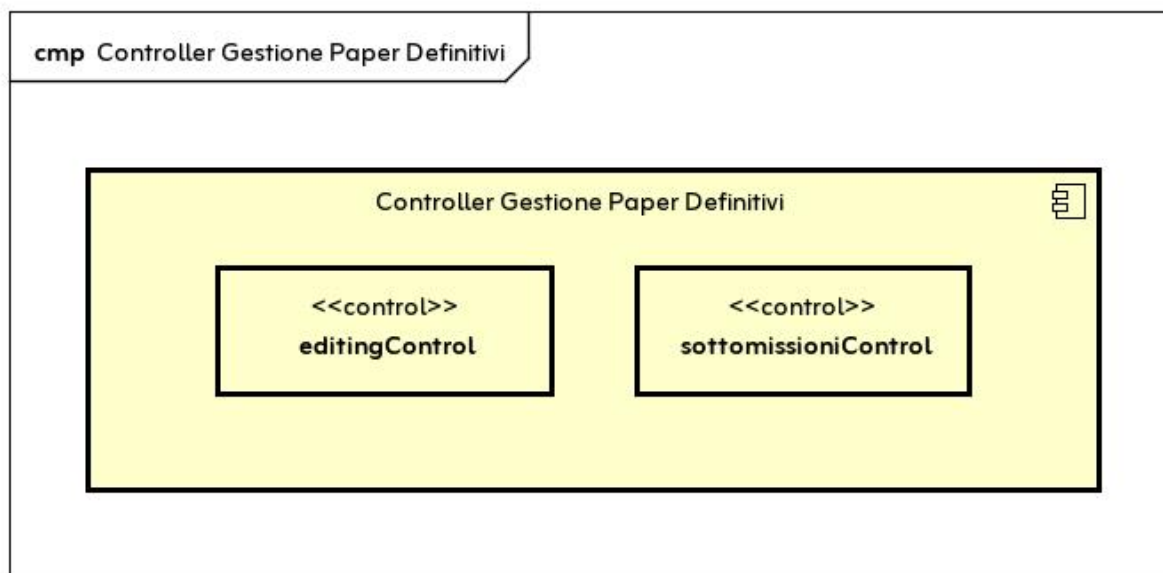
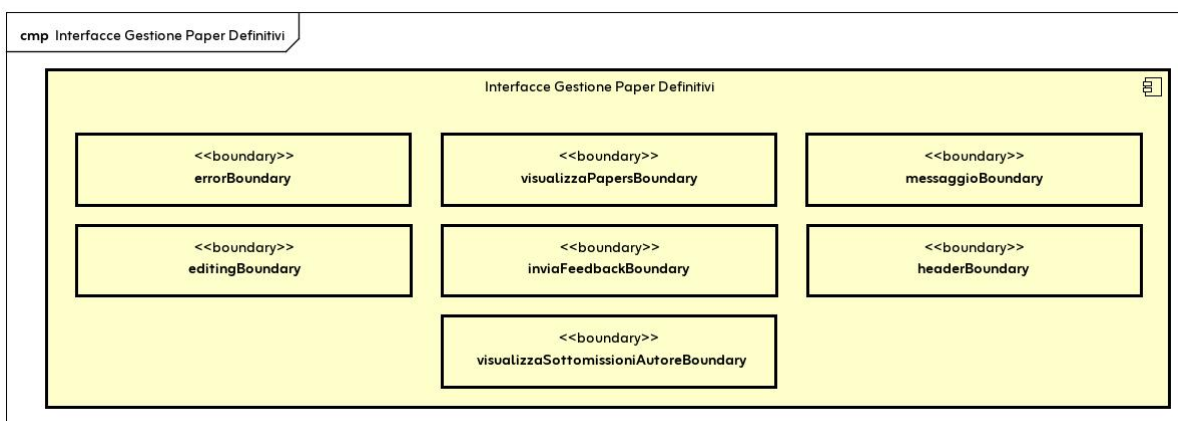
4.4.4 Gestione Revisioni



4.4.5 Presentazione Articolo



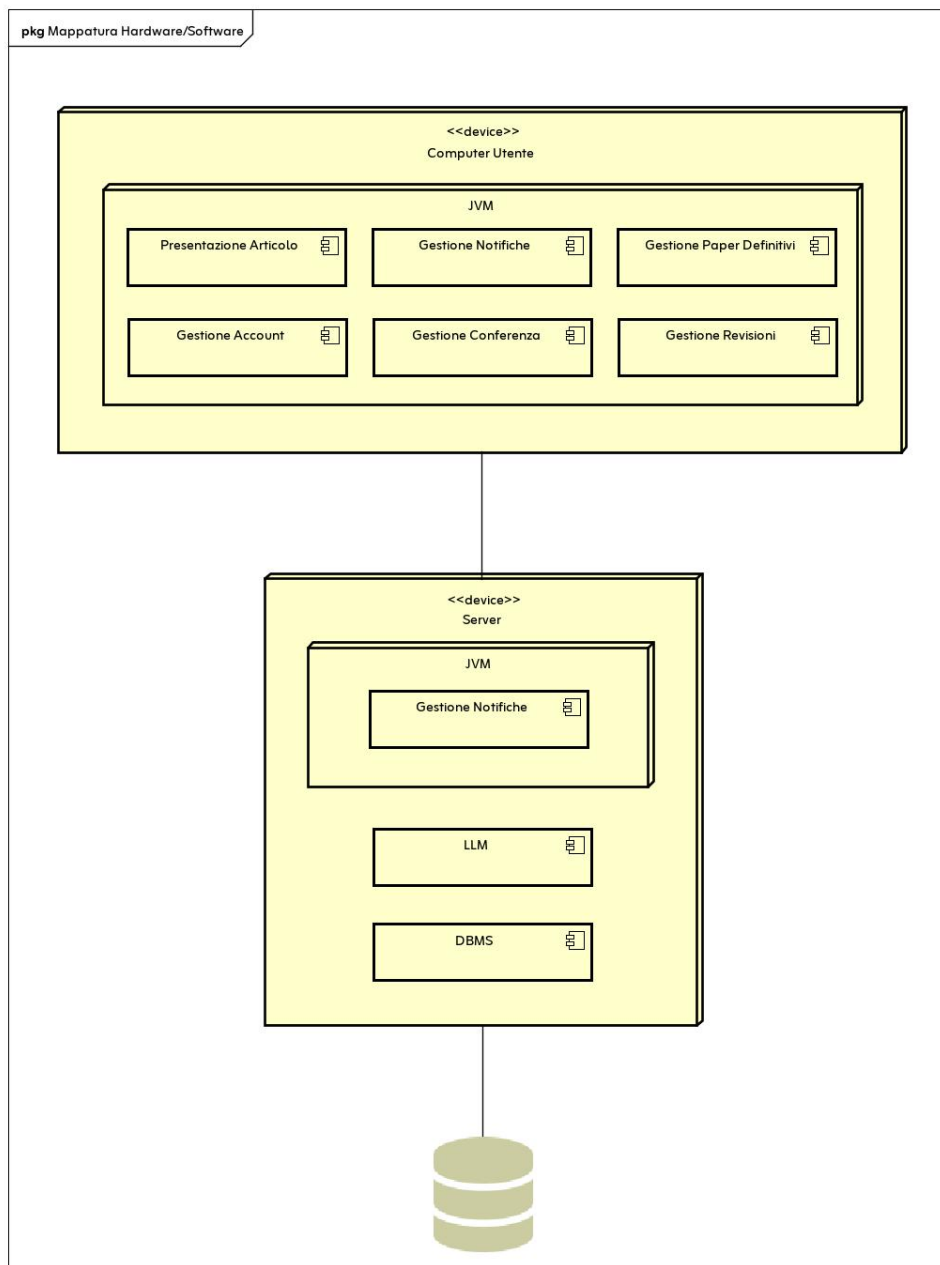
4.4.6 Gestione Paper Definitivi



4.5 Mappatura Hardware/Software

La mappatura del sistema è basata sul Modello architetturale Repository.

Il sistema prevede un nodo **“COMPUTER UTENTE”**, che ospita tutti i sottosistemi necessari al suo corretto funzionamento. Questo nodo è collegato al nodo centrale, denominato **“SERVER”**, che include il DBMS e le API necessari, rispettivamente, per recuperare e inserire informazioni nella sorgente dati e per comunicare con la LLM.

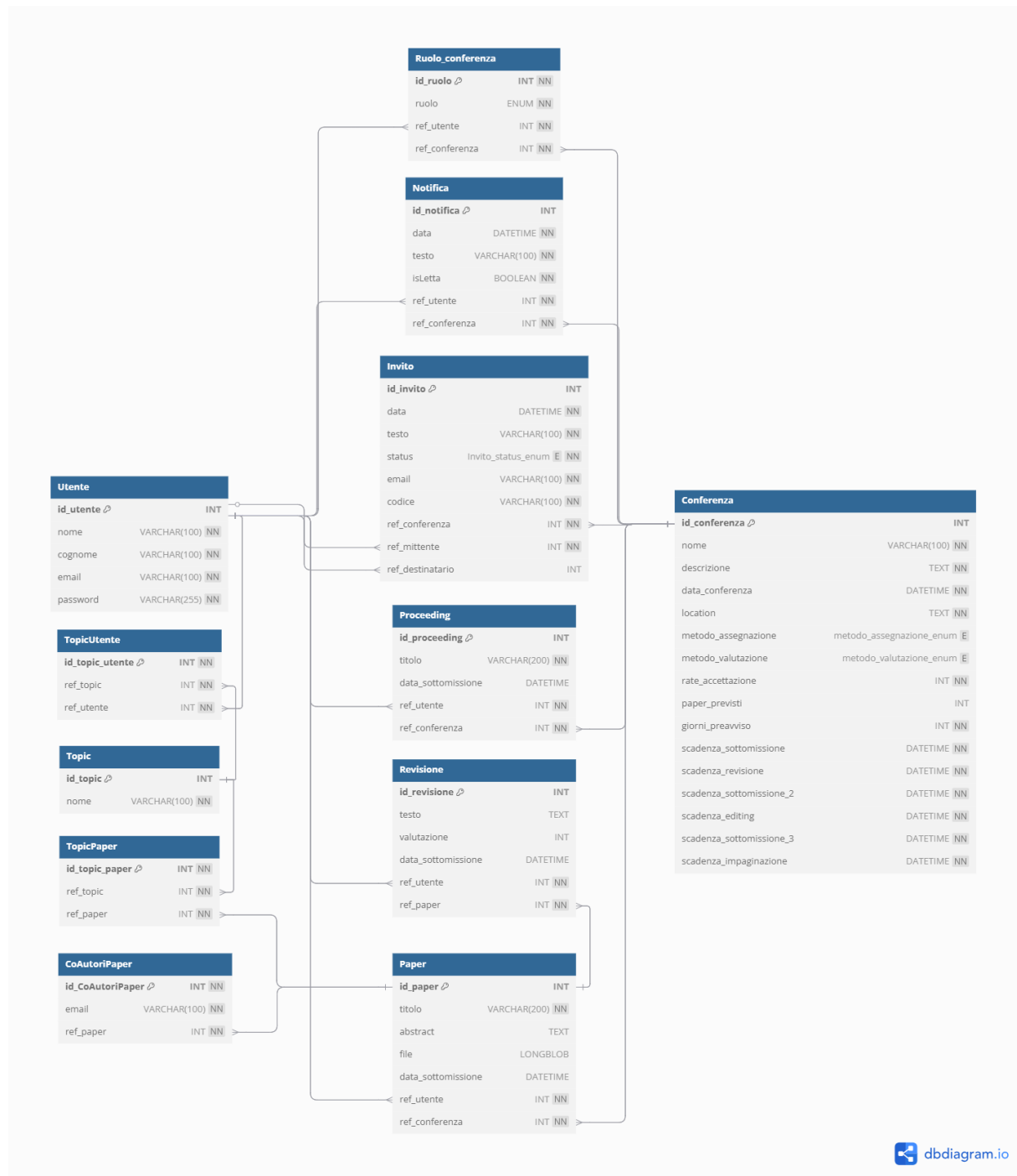


5 Gestione dei Dati Persistenti

Il database scelto per la gestione dei dati persistenti è un **database di tipo relazionale** e viene utilizzata un'interfaccia di tipo SQL, che consente di **rispettare le proprietà** definite dal protocollo **ACID** (Atomicity, Consistency, Isolation, Durability).

5.1 Schema E-R

Di seguito viene mostrato il modello Entità – Relazione del database gestito dal sistema:



5.2 Modello Relazionale

Di seguito viene mostrato il modello relazionale del database gestito dal sistema:

Utente (id_utente, nome, cognome, email, password)

PK: id_utente

Conferenza (id_conferenza, nome, descrizione, data_conferenza, location, metodo_assegnazione, metodo_valutazione, rate_accettazione, paper_previsti, giorni_preavviso, scadenza_sottomissione, scadenza_revisione, scadenza_sottomissione_2, scadenza_editing, scadenza_sottomissione_3, scadenza_impaginazione)

PK: id_conferenza

Notifica (id_notifica, data, testo, isLetta, ref_utente, ref_conferenza)

PK: id_notifica

FK: ref_utente references Utente(id_utente), ref_conferenza references Conferenza(id_conferenza)

Invito (id_invito, data, testo, status, email, codice, ref_conferenza, ref_mittente, ref_destinatario)

PK: id_invito

FK: ref_conferenza references Conferenza(id_conferenza), ref_mittente references Utente(id_utente), ref_destinatario references Utente(id_utente)

Topic (id_topic, nome)

PK: id_topic

Paper (id_paper, titolo, abstract, file, data_sottomissione, ref_utente, ref_conferenza)

PK: id_paper

FK: ref_utente references Utente(id_utente), ref_conferenza references Conferenza(id_conferenza)

Revisione (id_revisione, testo, valutazione, data_sottomissione, punti_forza, punti_debolezza, commento_chair, ref_utente, ref_paper)

PK: id_revisione

FK: ref_utente references Utente(id_utente), ref_paper references Paper(id_paper)

Proceeding (id_proceeding, titolo, data_sottomissione, ref_utente, ref_conferenza)

PK: id_proceeding

FK: ref_utente references Utente(id_utente), ref_conferenza references Conferenza(id_conferenza)

Ruolo_conferenza (id_ruolo, ruolo, ref_utente, ref_conferenza)

PK: id_chair

FK: ref_utente references Utente(id_utente), ref_conferenza references Conferenza(id_conferenza)

TopicUtente (id_topic_utente, ref_topic, ref_utente)

PK: id_topic_utente

FK: ref_topic references Topic(id_topic), ref_utente references Utente(id_utente)

TopicPaper (id_topic_paper, ref_topic, ref_paper)

PK: id_topic_paper

FK: ref_topic references Topic(id_topic), ref_paper references Paper(id_paper)

CoAutoriPaper (id_CoAutoriPaper, email, ref_paper)

PK: id_CoAutoriPaper

FK: ref_paper references Paper(id_paper)

5.3 Struttura delle tabelle

5.3.1 Utente

Nome attributo	Tipo	Vincoli	Descrizione
id_utente	integer(6)	PK, AutoIncrement	Codice identificativo univoco dell'utente
nome	varchar(100)	Not null	Nome dell'utente
cognome	varchar(100)	Not null	Cognome dell'utente
email	varchar(100)	Not null, UNIQUE	Email personale dell'utente
password	varchar(255)	Not null	Hash della password di accesso dell'utente

5.3.2 Notifica

Nome attributo	Tipo	Vincoli	Descrizione
id_notifica	integer(6)	PK, AutoIncrement	Codice identificativo della notifica
data	datetime	Not null	Data della notifica
testo	text	Not null	Testo della notifica
isLetta	boolean	Not null	Flag che indica se la notifica è stata letta
ref_utente	integer(6)	FK	Riferimento all'utente che riceve la notifica
ref_conferenza	integer(6)	FK	Riferimento alla conferenza a cui si riferisce la notifica

5.3.3 Topic

Nome attributo	Tipo	Vincoli	Descrizione
id.topic	integer(6)	PK, AutoIncrement	Codice identificativo del topic
nome	varchar(100)	Not null	Nome del topic

5.3.4 Conferenza

Nome attributo	Tipo	Vincoli	Descrizione
id_conferenza	integer(6)	PK, AutoIncrement	Codice identificativo univoco della conferenza
nome	varchar(100)	Not null	Nome della conferenza
descrizione	text	Not null	Descrizione della conferenza
data_conferenza	datetime	Not null	Data e ora della conferenza
location	text	Not null	Luogo della conferenza
metodo_assegnazione	enum('Broadcast', 'Topic')		Metodo di assegnazione automatico dei paper
metodo_valutazione	enum('2', '3', '4')		Metodo di valutazione dei paper
rate_accettazione	integer(6)	Not null	Rate di Accettazione dei paper
paper_previsti	integer(11)		Numero di paper previsti
giorni_preavviso	integer(11)	Not null, Default=3	Numero di giorni di preavviso per le scadenze
scadenza_sottomissione	datetime	Not null	Scadenza per la sottomissione dei paper
scadenza_revisione	datetime	Not null	Scadenza per la revisione dei paper
scadenza_sottomissione_2	datetime	Not null	Scadenza seconda sottomissione dei paper
scadenza_editing	datetime	Not null	Scadenza editing finale dei paper
scadenza_sottomissione_3	datetime	Not null	Scadenza terza sottomissione dei paper
scadenza_impaginazione	datetime	Not null	Scadenza impaginazione finale dei paper

5.3.5 Invito

Nome attributo	Tipo	Vincoli	Descrizione
id_invito	integer(6)	PK, AutoIncrement	Codice identificativo dell'invito
data	datetime	Not null	Data della scadenza dell'invito
testo	varchar(100)	Not null	Testo dell'invito
status	enum('Inviato', 'Accettato', 'Rifiutato')	Not null	Stato dell'invito
email	text	Not null	Email del destinatario
codice	varchar(100)	Not null, UNIQUE	Codice identificativo unico dell'invito
ref_conferenza	integer(6)	FK	Riferimento alla conferenza
ref_mittente	integer(6)	FK	Riferimento all'utente che invia l'invito
ref_destinatario	integer(6)	FK	Riferimento all'utente destinatario dell'invito

5.3.6 Paper

Nome attributo	Tipo	Vincoli	Descrizione
id_paper	integer(6)	PK, AutoIncrement	Codice identificativo del paper
titolo	varchar(200)	Not null	Titolo del paper
abstract	text		Abstract del paper
file	longblob		PDF del Paper
data_sottomissione	datetime		Data di sottomissione del paper
ref_utente	integer(6)	FK	Riferimento all'utente che ha sottomesso il paper
ref_conferenza	integer(6)	FK	Riferimento alla conferenza a cui è stato sottomesso il paper

5.3.7 Revisione

Nome attributo	Tipo	Vincoli	Descrizione
id_revisione	integer(6)	PK, AutoIncrement	Codice identificativo della revisione
testo	text		Testo della revisione
valutazione	integer(11)		Valutazione del paper
data_sottomissione	datetime		Data di sottomissione della revisione
punti_forza	text		Descrizione punti di forza del paper revisionato
punti_debolezza	text		Descrizione punti di debolezza del paper revisionato
commento_chair	text		Commento privato da inviare al chair in caso di problemi
ref_utente	integer(6)	FK	Riferimento all'utente che ha effettuato la revisione
ref_paper	integer(6)	FK	Riferimento al paper oggetto della revisione

5.3.8 Proceeding

Nome attributo	Tipo	Vincoli	Descrizione
id_proceeding	integer(6)	PK, AutoIncrement	Codice identificativo del proceeding
titolo	varchar(200)	Not null	Titolo del proceeding
data_sottomissione	datetime		Data di sottomissione del proceeding
ref_utente	integer(6)	FK	Riferimento all'utente che ha sottomesso il proceeding
ref_conferenza	integer(6)	FK	Riferimento alla conferenza a cui è stato sottomesso il proceeding

5.3.9 Ruolo_conferenza

Nome attributo	Tipo	Vincoli	Descrizione
id_ruolo	integer(6)	PK, AutoIncrement	Codice identificativo del ruolo di un Utente in una determinata Conferenza
ruolo	enum("Chair", "Revisore", "Sottorevisore", "Autore", "Editor")	Not null	Ruolo dell'utente in quella specifica conferenza
ref_utente	integer(6)	FK	Riferimento all'utente a cui è riferito il ruolo
ref_conferenza	integer(6)	FK	Riferimento alla conferenza a cui l'utente fa parte

5.3.10 TopicUtente

Nome attributo	Tipo	Vincoli	Descrizione
id_topic_utente	integer(6)	PK, AutoIncrement	Codice identificativo dell'associazione topic-utente
ref_topic	integer(6)	FK	Riferimento al topic associato
ref_utente	integer(6)	FK	Riferimento all'utente associato al topic

5.3.11 CoAutoriPaper

Nome attributo	Tipo	Vincoli	Descrizione
id_CoAutoriPaper	integer(6)	PK, AutoIncrement	Codice identificativo del Coautore del Paper
email	varchar(100)	Not null	Email del Coautore
ref_paper	integer(6)	FK	Riferimento al paper associato al Coautore

6 Copyright e diritto d'autore

La presente **documentazione** è protetta dalle **leggi sul diritto d'autore**. Nessuna parte di questo documento può essere riprodotta, distribuita o trasmessa in alcuna forma o con alcun mezzo, elettronico o meccanico, inclusa la fotocopia, la registrazione o altri sistemi di memorizzazione o recupero di informazioni, senza il **previo consenso scritto degli autori**.



PaperReview © 2025

Leonardo Giovanni Cozzella

Diego Corona

Luca Gaston

Samuele D'Amico