



UNIVERSITÀ DEGLI STUDI DI PALERMO
FACOLTÀ DI INGEGNERIA

LAUREA TRIENNALE IN INGEGNERIA INFORMATICA



PAPERPILOT

AUTORI

Mirko Stefano Cirivello

Maria Chiara Ciuffoli

Flavio Giarrusso

Alessia Pizzuto

DOCENTE

Dott.sa Seidita Valeria

1. Obiettivo del Sistema.....	2
1.2. Architettura Attuale	2
2. Architettura Software Proposta	2
2.1 Panoramica.....	2
2.2 Requisiti minimi per l'utilizzo del software proposto.....	2
2.3 Decomposizione in Sottosistemi.	3
2.4 Suddivisione degli Oggetti all'interno dei Sottosistemi e delle Componenti	5
2.5 Mappatura Hardware/Software	9
3. Gestione dei Dati Persistenti	10
3.1 Modello E-R (Entity-Relationship)	10
3.2 Modello Relazionale	11
3.3 Struttura delle Tabelle.....	13

1. Obiettivo del Sistema

- L'obiettivo del nostro sistema è la realizzazione di un software per gestire le conferenze cms. Il fine è pertanto quello di agevolare la comunicazione tra l'utente e il sistema in modo da rendere l'interfaccia semplice e facilmente accessibile a chiunque voglia utilizzarla. Gli utenti che possono utilizzare le funzioni del sistema sono: Chair, Revisori, Sotto revisori, Editori, Autori e Membri del pc

1.2. Architettura Attuale

Supponiamo che non esista nessuna soluzione software usata attualmente per svolgere le funzioni che il sistema proposto può effettuare.

2. Architettura Software Proposta

2.1 Panoramica

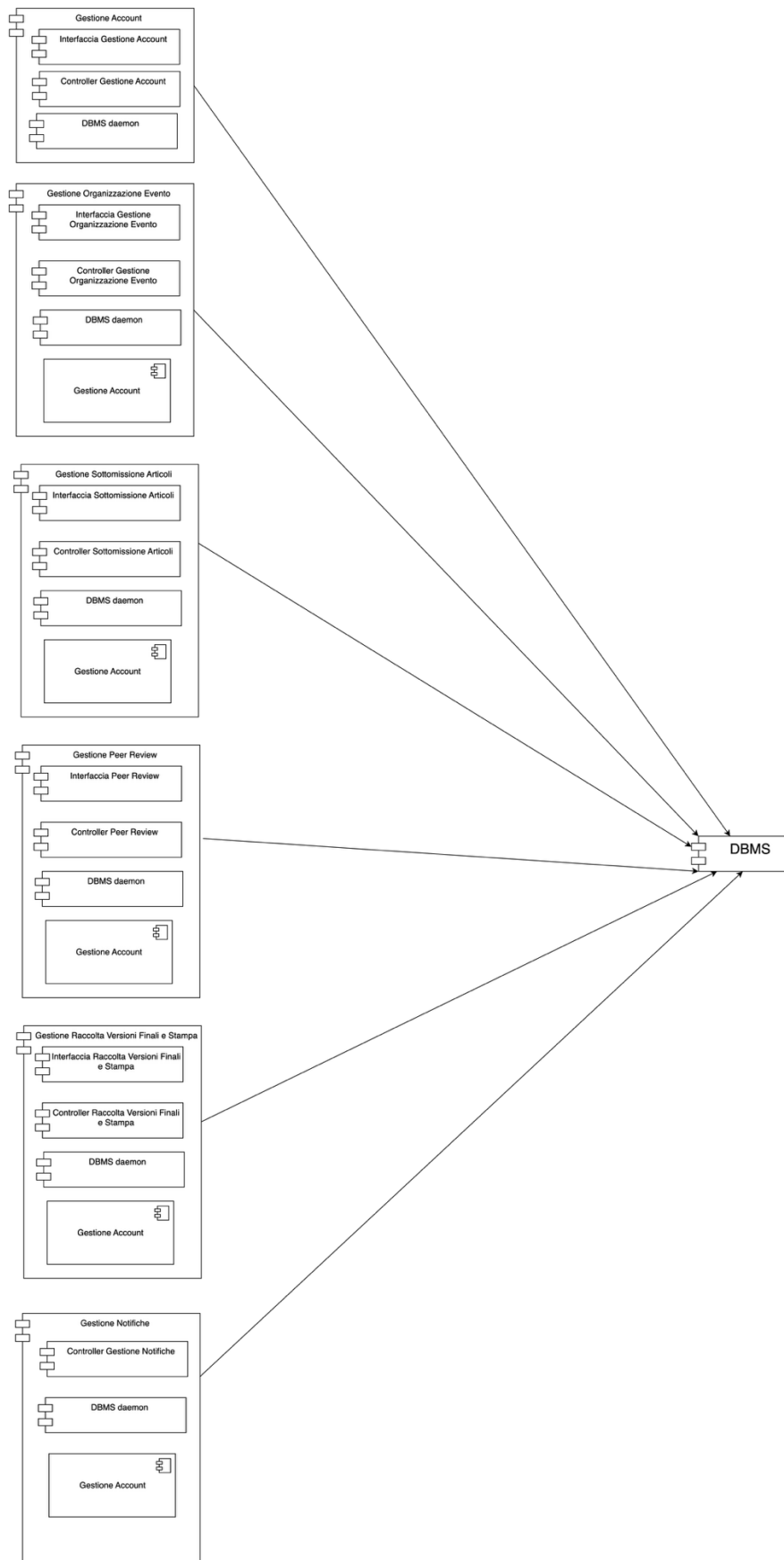
Per la realizzazione del software **PaperPilot** è stata utilizzata un'architettura di tipo **Repository**, che garantisce il disaccoppiamento dei sottosistemi, i quali possono comunicare solo con il sottosistema di Storage, ma non tra loro. A loro volta i sottosistemi sono composti da un'interfaccia utente che comunica solo con il controllore sottostante, che contiene la logica del programma e che gestisce le richieste da rivolgere al sottosistema DBMSDaemon, che gestisce le comunicazioni al nodo di Storage. L'interfaccia utente, il controllore e il DBMSDaemon di ogni sottosistema risiedono sullo stesso nodo, mentre il sottosistema di Storage risiede su un nodo diverso.

2.2 Requisiti minimi per l'utilizzo del software proposto

Per funzionare correttamente, il Software necessita di:

Connessione a Internet stabile per permettere le comunicazioni ai DBMS e per permettere al Sistema di inviare le Mail necessarie.

2.3 Decomposizione in Sottosistemi.

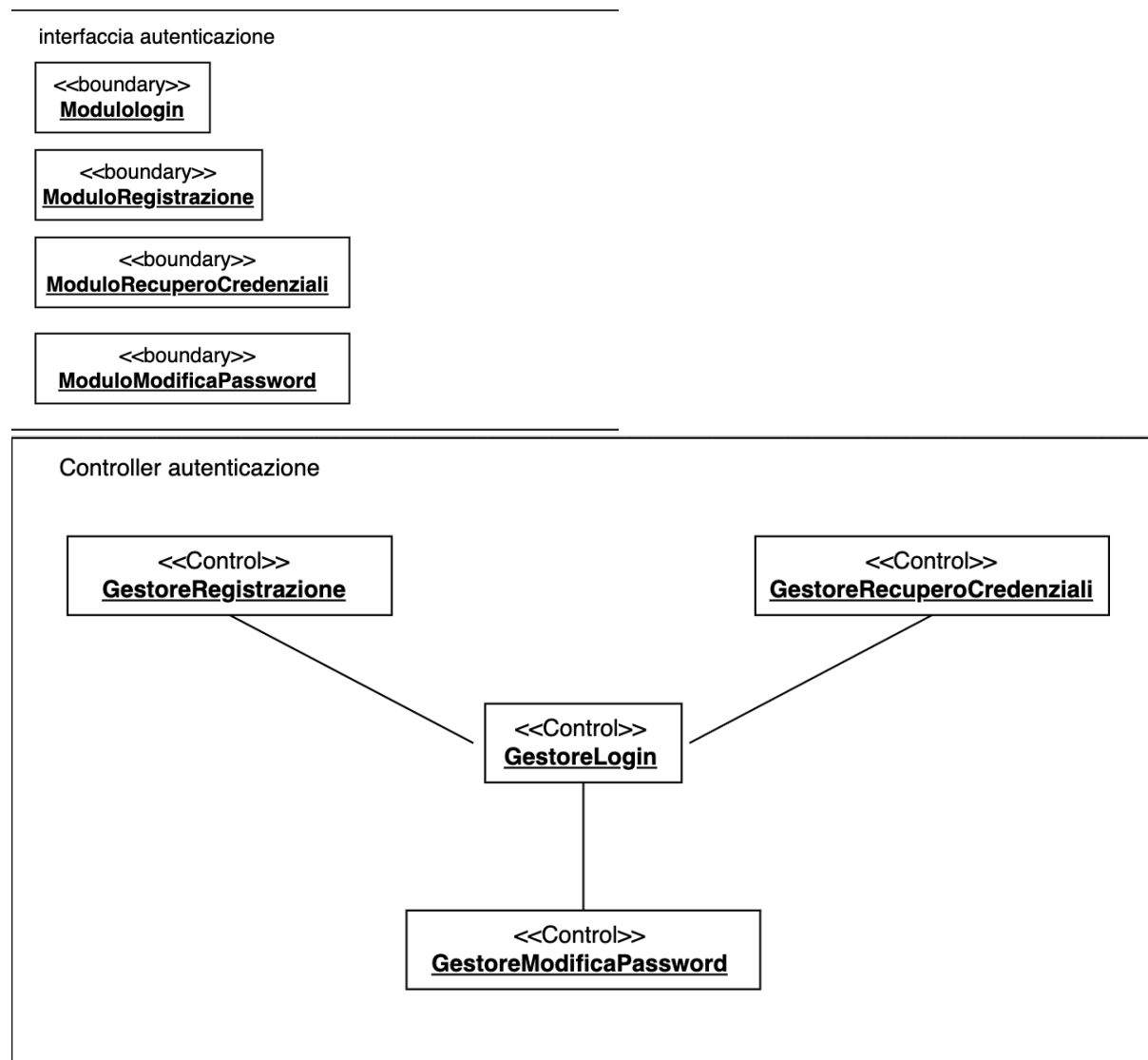


Il **sottosistema Gestione Account** agli utenti di autenticarsi, recuperare le credenziali e modificare i dati del proprio profilo, questo sottosistema è presente in ogni altro sottosistema.. Il **sottosistema Gestione organizzazione evento** permette di gestire tutte le funzionalità che riguardano la gestione della conferenza. Il Chair può creare un nuovo evento e invita tramite e-mail degli esperti per essere un revisore; inoltre, definisce un numero minimo di revisori da assegnare automaticamente agli articoli e/o definisce il numero massimo di articoli da assegnare ad un revisore. Può anche definire il range simmetrico di valutazione degli articoli (3 o 5) e le varie date di scadenza per la presentazione dei paper, la loro revisione, l'accettazione dei paper, l'invio della versione finale. I revisori verranno a loro volta notificati dell'invito tramite una mail. Nel **sottosistema gestione sottomissione articoli** sono presenti tutte le funzionalità relative all'invio dei documenti (che conterranno gli articoli) e alla relativa assegnazione di quest'ultimi a specifici Revisori. In seguito ad una scadenza temporale gli Autori inviano i documenti, che verranno salvati nel DBMS, e saranno informati dell'avvenuta ricezione. Il sistema assegna automaticamente un numero di revisori per ogni articolo, i Revisori possono essere successivamente modificati dal Chair; in particolare, può decidere di aggiungere o rimuovere specifici Revisori per specifici articoli in base ad alcuni criteri come la propria preferenza personale oppure le competenze possedute dal revisore riguardante l'argomento dell'articolo. Un revisore ha anche la possibilità di contattare dei sotto-revisori per esaminare uno o più articoli: l'obiettivo è far sì che un revisore o sotto-revisore abbia la possibilità di accettare o rifiutare l'articolo a cui è stato assegnato, così che il sistema comunichi al chair via e-mail della scelta intrapresa. Nel **sottosistema gestione peer review** sono presenti tutte le funzionalità riguardanti la fase di revisione dell'articolo. Una revisione dovrà esser pronta entro una data prefissata dal Chair, la quale conterrà alcune informazioni rilevanti come: un commento e una valutazione numerica (feedback) Le recensioni degli articoli verranno utilizzate per decretare l'accettazione o il rifiuto degli stessi; in particolare, il sistema dovrà generare una graduatoria entro una specifica data di scadenza, la quale conterrà la media aritmetica dei voti assegnati da parte dei Revisori per ciascun articolo: una volta realizzata tale graduatoria, verrà comunicata mediante notifica al Chair. Quest'ultimo avrà il compito di visualizzare la graduatoria e di stabilire quali sono gli articoli da accettare o rifiutare: gli autori degli articoli accettati riceveranno una notifica da parte del sistema. Nel **sottosistema gestione raccolta versioni finali e stampa** sono presenti tutte le funzionalità atte alla raccolta degli articoli nella loro versione finale e della loro stampa. Gli autori invieranno una versione definitiva del loro articolo entro una certa data di scadenza. Il sistema, nel caso in cui l'autore non abbia presentato l'articolo finale entro i tempi stabiliti, deve notificare almeno uno o due giorni antecedenti rispetto alla data di scadenza l'autore riguardo la consegna dell'articolo. Raggiunta la scadenza, il sistema raccoglie gli articoli caricati dagli autori

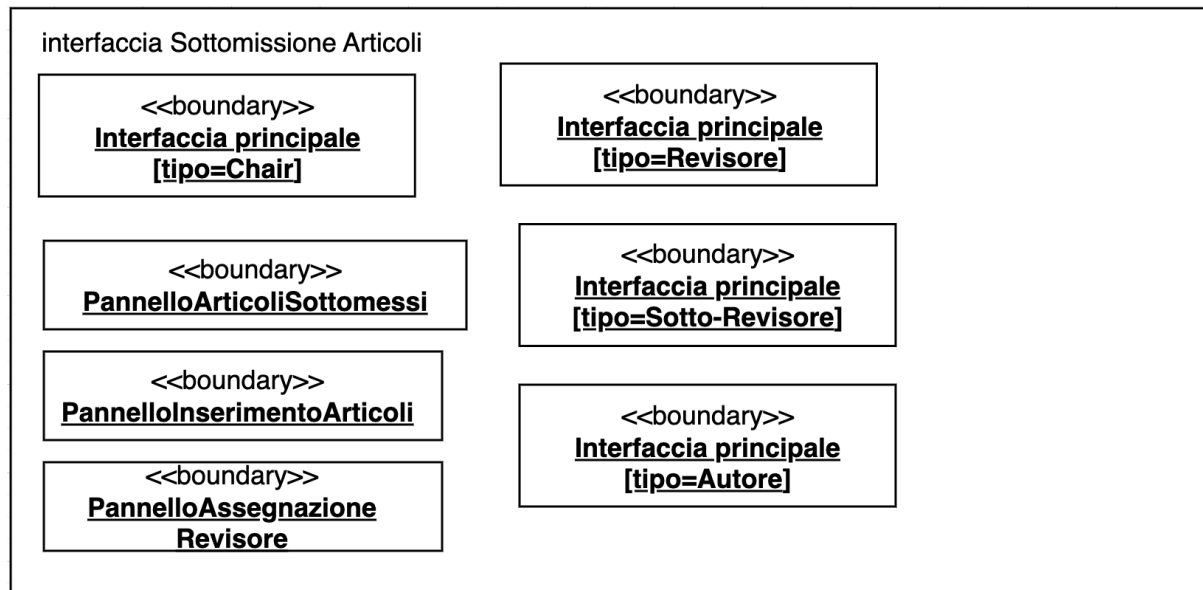
nel DBMS e li invierà all'editore, che ha la possibilità di visualizzare la lista degli articoli finali e di consultarne il contenuto. Nel **sottosistema Gestione Notifiche** vengono gestite tutte le notifiche che il sistema invia agli utenti.

2.4 Suddivisione degli Oggetti all'interno dei Sottosistemi e delle Componenti

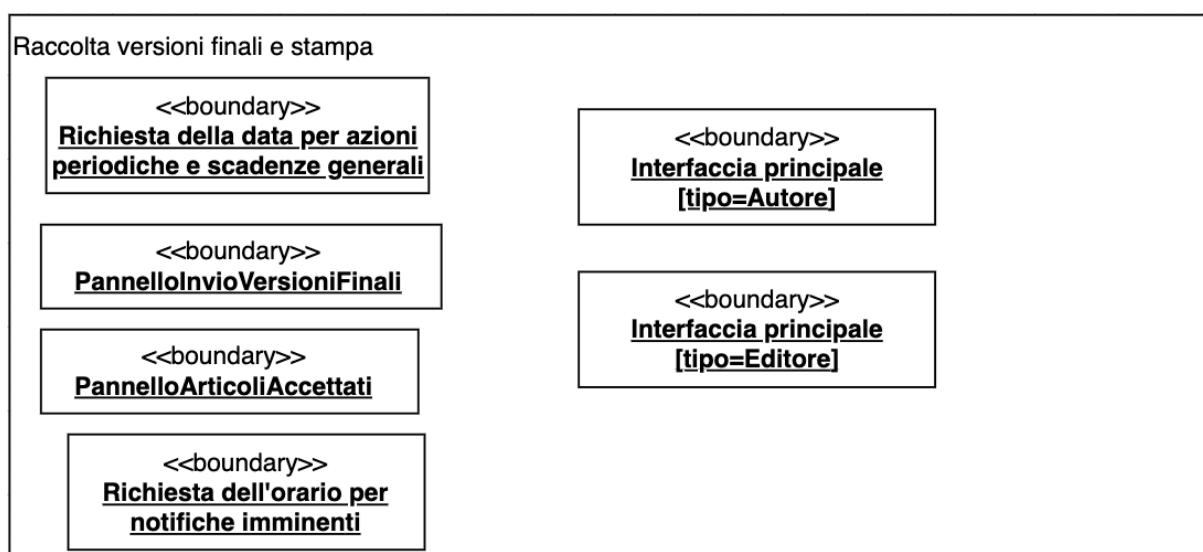
Modulo Login



Sottomissione Articoli



Versioni Finali e stampa



Controller Versioni Finali e Stampa

<<Control>>
GestoreVersioniFinaliControl

<<Control>>
GestorePubblicazioneControl

Peer Review

interfaccia Peer review

<<boundary>>
ModuloValutazioneArticolo

<<boundary>>
PannelloFeedbackRevisioni

<<boundary>>
VisualizzatoreArticoli

<<boundary>>
PannelloInviaNotificheAll'Autore

<<boundary>>
Interfaccia principale
[tipo=Chair]

<<boundary>>
Interfaccia principale
[tipo=Revisore]

<<boundary>>
Interfaccia principale
[tipo=Sotto-Revisore]

<<boundary>>
Richiesta dell'orario per
notifiche imminenti

<<boundary>>
Richiesta della data per azioni
periodiche e scadenze generali

Controller Peer Review

<<Control>>
GestoreRecensioni

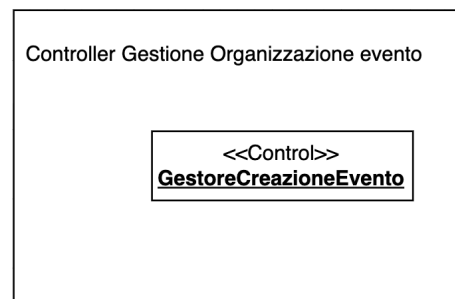
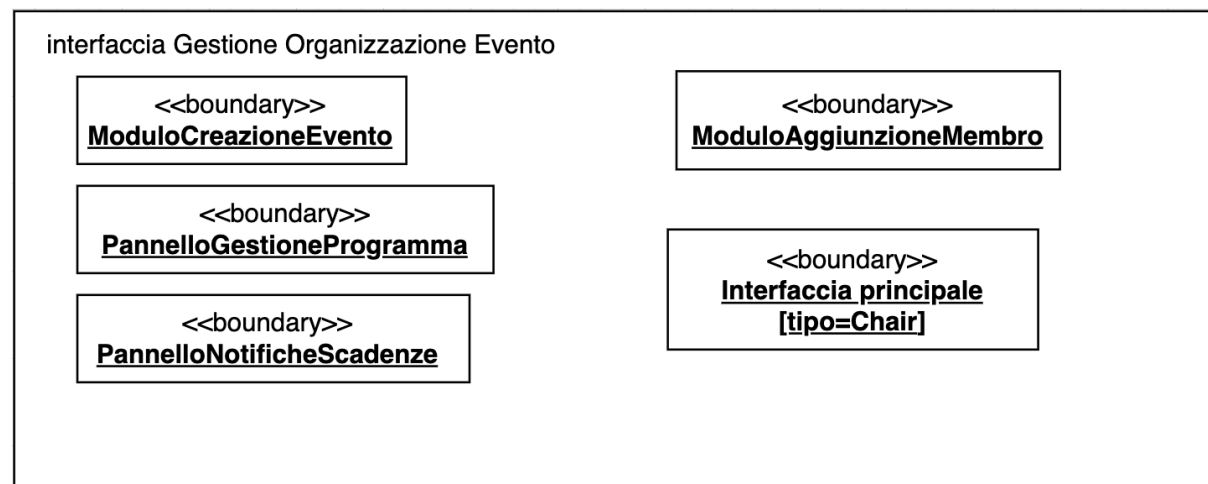
<<Control>>
GestoreNotificaEsitoArticolo

<<Control>>
GestoreTempisticaPeerReview

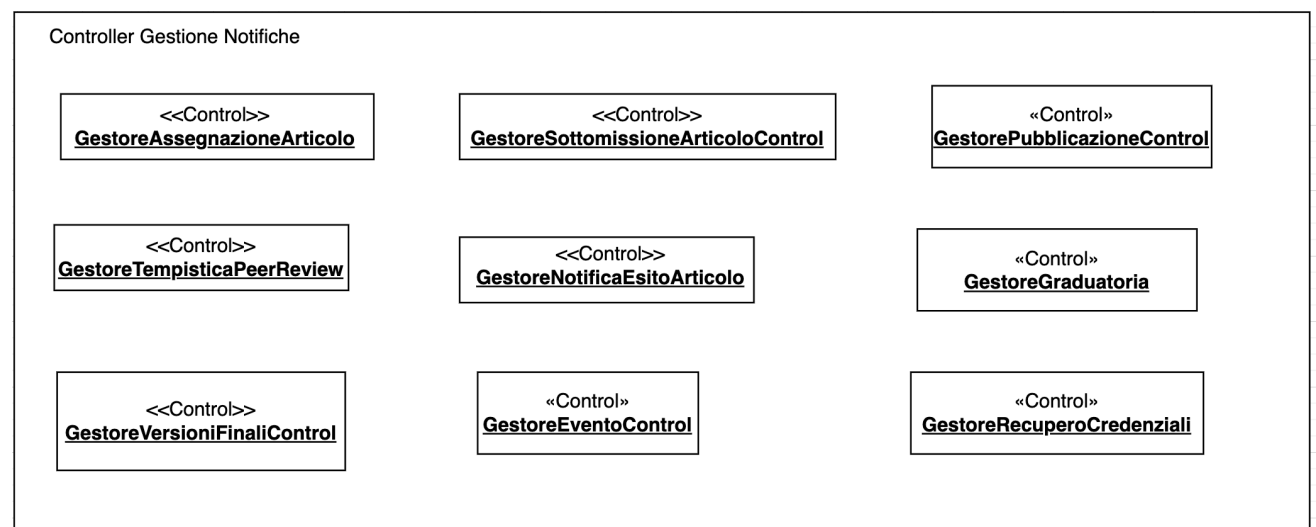
<<Control>>
GestoreGraduatoria

<<Control>>
GestoreInvioRevisioni

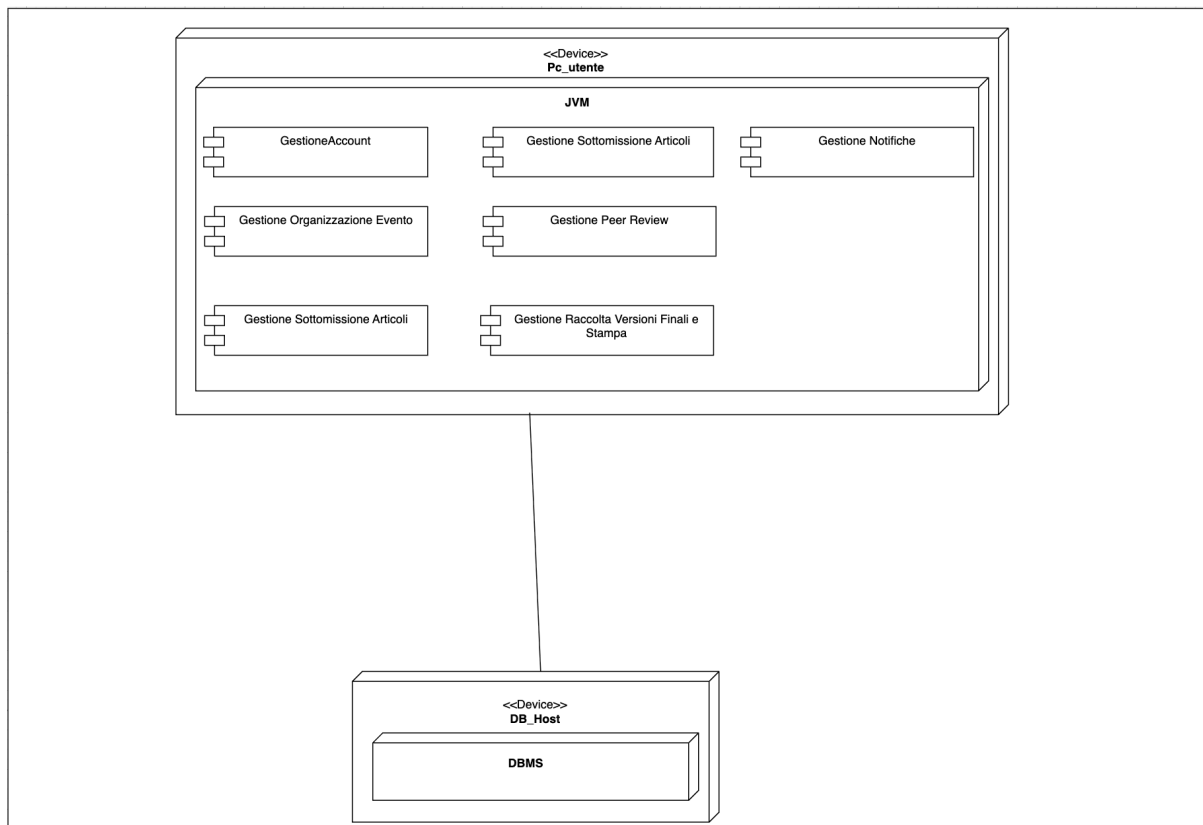
Organizzazione evento



Gestione Notifiche



2.5 Mappatura Hardware/Software



Per l'ambiente di produzione di PaperPilot, si prevede una configurazione hardware e software. La mappatura è stata effettuata sulla base del modello architetturale **Repository**.

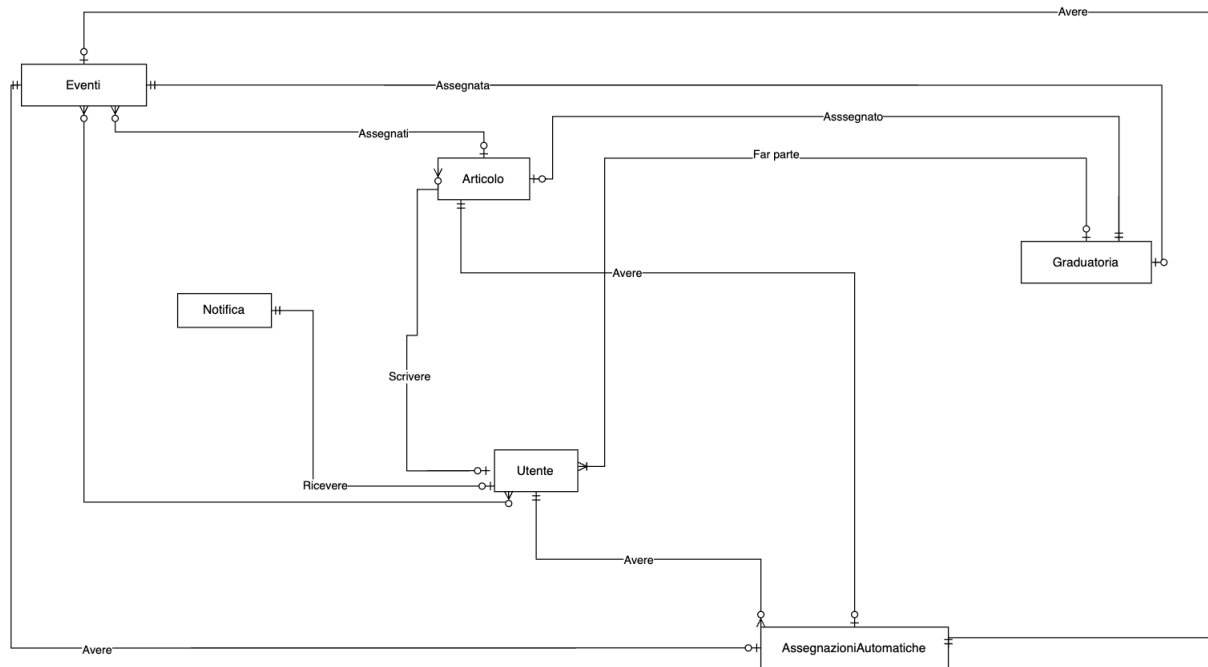
Sul nodo **PC_Utente** è integrato il modulo di **Gestione account** necessario per il corretto funzionamento del sistema, sono inoltre associati i singoli pacchetti software specifici per il ruolo assegnato (ad esempio, GestioneOrganizzazioneEvento, GestioneSottomissione articoli e Gestione Notifiche), che consentono all'utente di svolgere le proprie funzionalità.

Il nodo **Db_Host** contiene il **DBMS** necessario nella memorizzazione e distribuzione dei dati.

3. Gestione dei Dati Persistenti

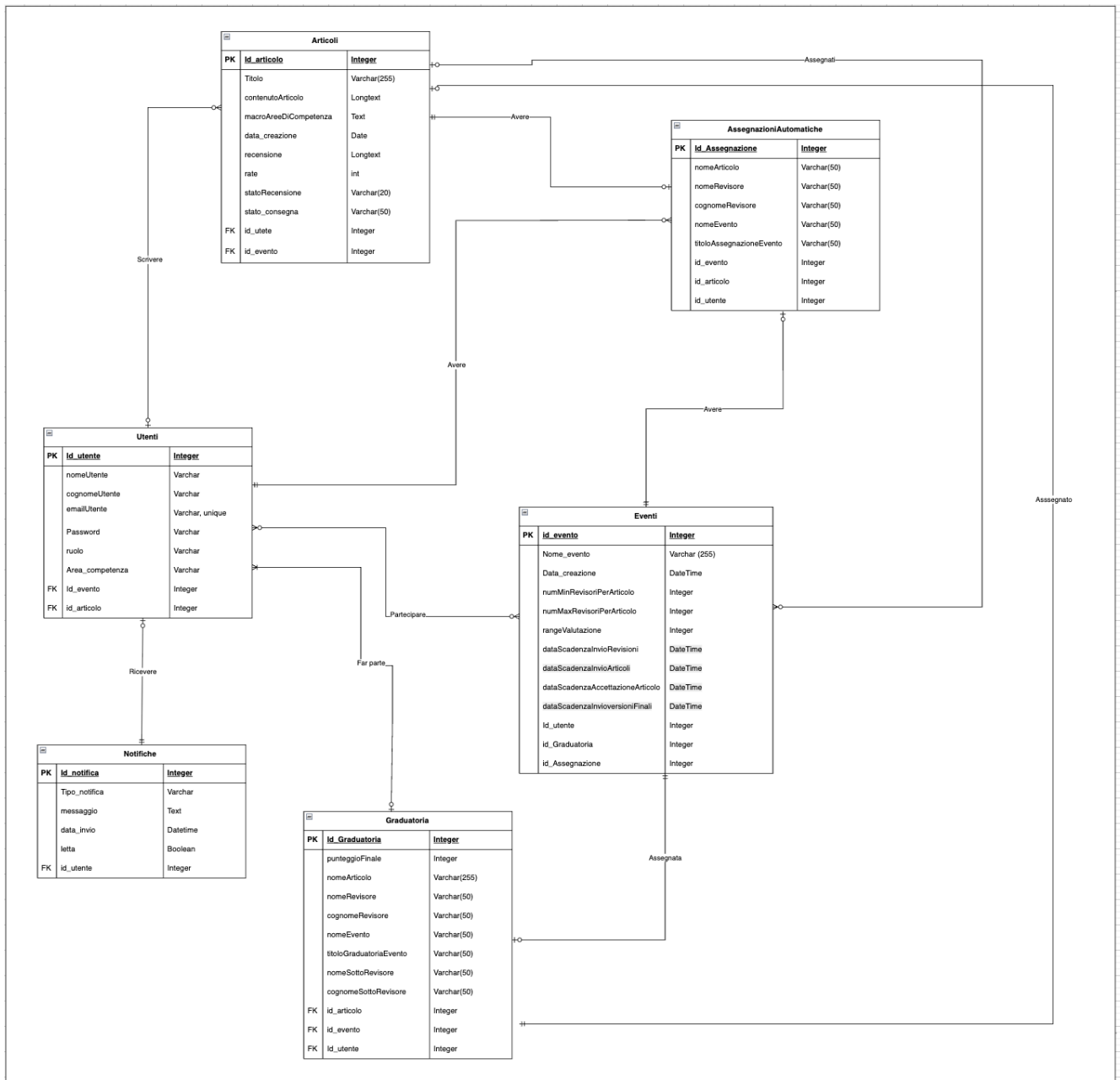
3.1 Modello E-R (Entity-Relationship)

Il Modello E-R definisce le entità principali del sistema PaperPilot e le relazioni tra di esse



3.2 Modello Relazionale

Il Modello Relazionale è la traduzione del Modello E-R in tabelle, chiavi primarie (PK) e chiavi esterne (FK).



Modello Relazionale

Utenti (id_utente, nomeUtente, cognomeUtente, emailUtente, password, ruolo)

- PK: id_utente

Articoli (id_articolo, titolo, contenutoArticolo, macroAreeDiCompetenza, recensione, rate, statoRecensione data_creazione, stato_consegna, id_utente)

- PK: id_articolo
- FK: id_utente references Utenti (id_utente)

Notifiche (id_notifica, tipo_notifica, messaggio, data_invio, letta, id_destinatario)

- PK: id_notifica
- FK: id_destinatario references Utenti (id_utente)

Graduatoria (Id_Graduatoria, punteggioFinale, nomeArticolo, nomeRevisore, cognomeRevisore, nomeEvento, titoloGraduatoriaEvento, nomeSottoRevisore, cognomeSottoRevisore, id_articolo, id_evento, id_utente)

- PK: Id_Graduatoria
- FK: id_articolo references Articoli (id_articolo)
- FK: id_utente references Utenti (id_utente)
- FK: id_evento references Eventi (id_evento)

Eventi (id_evento, Nome_evento, Data_creazione, numMinRevisoriPerArticolo, numMaxArticoliPerRevisore, rangeValutazione, dataScadenzaInvioArticoli, dataScadenzaInvioRevisioni, dataScadenzaAccettazioneArticolo, dataScadenzaInvioVersioniFinali, id_utente, Id_Graduatoria)

- PK: id_evento
- FK: id_utente references Utenti (id_utente)
- FK: Id_Graduatoria references Graduatoria (Id_Graduatoria)

AssegnazioniAutomatiche (id_Assignazione, nomeArticolo, nomeRevisore, cognomeRevisore, nomeEvento, titoloAssegnazioneEvento, id_evento, id_articolo, id_utente)

- PK: id_Assignazione
- FK: id_evento references Eventi (id_evento)
- FK: id_articolo references Articoli (id_articolo)
- FK: id_utente references Utenti (id_utente)

I vincoli di tupla presenti sono i seguenti:

- La Tabella **Articoli** è collegata alla Tabella **Utenti** tramite la colonna **id_utente**.
- La Tabella **Articoli** è collegata alla Tabella **Eventi** tramite la colonna **id_evento**.
- La Tabella **Notifiche** è collegata alla Tabella **Utenti** tramite la colonna **id_destinatario**.
- La Tabella **Graduatoria** è collegata alla Tabella **Articoli** tramite la colonna **id_articolo**.
- La Tabella **Graduatoria** è collegata alla Tabella **Utenti** tramite la colonna **id_utente**.
- La Tabella **Graduatoria** è collegata alla Tabella **Eventi** tramite la colonna **id_evento**.
- La Tabella **Eventi** è collegata alla Tabella **Utenti** tramite la colonna **id_utente**.
- La Tabella **Eventi** è collegata alla Tabella **Graduatoria** tramite la colonna **Id_Graduatoria**.
- La Tabella **AssegnazioniAutomatiche** è collegata alla Tabella **Eventi** tramite la colonna **id_evento**.
- La Tabella **AssegnazioniAutomatiche** è collegata alla Tabella **Articoli** tramite la colonna **id_articolo**.
- La Tabella **AssegnazioniAutomatiche** è collegata alla Tabella **Utenti** tramite la colonna **id_utente**.

3.3 Struttura delle Tabelle

Di seguito la definizione dettagliata delle tabelle come implementate nel database.

Tabella: Utenti

Nome Colonna	Tipo Dati	Vincoli	Descrizione
id_utente	INTEGER	PK, AUTO_INCREMENT	Identificativo univoco dell'utente.
nomeUtente	VARCHAR(255)	NOT NULL	Nome dell'utente.
cognomeUtente	VARCHAR(255)	NOT NULL	Cognome dell'utente.
emailUtente	VARCHAR(255)	NOT NULL, UNIQUE	Indirizzo email dell'utente, utilizzato anche per l'accesso.
password	VARCHAR(255)	NOT NULL	Hash della password dell'utente per sicurezza.
ruolo	VARCHAR(50)	NOT NULL, ENUM('autore', 'editore')	Ruolo dell'utente all'interno del sistema.
Id_evento	INTEGER	Fk, NOT NULL, UNIQUE	Identificativo univoco dell'evento.
id_articolo	INTEGER	FK, NOT NULL	Identificativo univoco dell'articolo.
Area_competenza	VARCHAR(255)	NOT NULL	Ruolo dell'utente all'interno della sua area.

Tabella: Articoli

Nome Colonna	Tipo Dati	Vincoli	Descrizione
id_articolo	INTEGER	PK, AUTO_INCREMENT	Identificativo univoco dell'articolo.

titolo	VARCHAR(255)	NOT NULL	Titolo dell'articolo.
recensione	LONGTEXT	NOT NULL	Contenuto della recensione dell'articolo
rate	INTEGER	NOT NULL	Valore di valutazione per l'articolo
statoRecensione	VARCHAR(20)	NOT NULL	Stato della recensione(esempio: incompleta, completata)
contenutoArticolo	LONGTEXT	NOT NULL	Contenuto dell'articolo.
macroAreeDiCompetenza	TEXT	NULLABLE	Breve descrizione della disciplina tratta nell'articolo.
data_creazione	DATE	NOT NULL	Data di creazione dell'articolo nel sistema.
stato_consegna	VARCHAR(50)	NOT NULL, DEFAULT 'Da Inviare'	Stato attuale Dell articolo (es. 'Inviato, Notificato').
id_utente	INTEGER	NOT NULL, FK(Utenti)	Identificativo dell'utente.
id_evento	INTEGER	FK, AUTO_INCREMENT	Chiave primaria univoca per ogni evento.

Tabella: Notifiche

Nome Colonna	Tipo Dati	Vincoli	Descrizione
id_notifica	INTEGER	PK, AUTO_INCREMENT	Identificativo univoco della notifica.
tipo_notifica	VARCHAR(50)	NOT NULL, ENUM('Scadenza', 'Conferma Invio', 'Promemoria', 'Info')	Categoria della notifica.
messaggio	TEXT	NOT NULL	Contenuto testuale della notifica.

data_invio	DATETIME	NOT NULL	Data e ora in cui la notifica è stata generata/inviata.
letta	BOOLEAN	NOT NULL, DEFAULT FALSE	Indica se la notifica è stata letta dall'utente.
id_destinatario	INTEGER	NOT NULL, FK(Utenti)	Utente a cui è destinata la notifica.

Tabella: Graduatoria

Nome Colonna	Tipo Dati	Vincoli	Descrizione
Id_Graduatoria	INTEGER	PK, AUTO_INCREMENT	Identificativo univoco per ogni voce di graduatoria.
punteggioFinale	INTEGER	NOT NULL	Il valore numerico che determina la posizione dell'articolo nella graduatoria.
nomeArticolo	VARCHAR(255)	NOT NULL	Titolo dell'articolo.
nomeRevisore	VARCHAR(50)	NOT NULL	Nome del revisore assegnato.
cognomeRevisore	VARCHAR(50)	NOT NULL	Cognome del revisore assegnato.
nomeEvento	VARCHAR(50)	NOT NULL	Nome dell'evento.
titoloGraduatoriaEvento	VARCHAR(50)	NOT NULL	Titolo della graduatoria in base all'evento assegnato.
nomeSottoRevisore	VARCHAR(50)	NOT NULL	Nome del sotto revisore assegnato.
cognomeSottoRevisore	VARCHAR(50)	NOT NULL	Cognome del sotto revisore assegnato.
id_articolo	INTEGER	NOT NULL, FK	Identificativo dell'articolo a cui si riferisce questa voce di graduatoria.

id_evento	INTEGER	FK, AUTO_INCREMENT	Chiave primaria univoca per ogni evento.
id_utente	INTEGER	FK, AUTO_INCREMENT	Identificativo univoco dell'utente.

Tabella: Eventi

Nome Colonna	Tipo Dati	Vincoli	Descrizione
id_evento	INTEGER	PK, AUTO_INCREMENT	Chiave primaria univoca per ogni evento.
Nome_evento	VARCHAR(255)	NOT NULL	Il nome o titolo dell'evento
Data_creazione	DATETIME	NOT NULL	La data dell'evento.
numMinRevisoriPerArticolo	INTEGER	NOT NULL	Valore che delimita il valore minimo di revisori per articolo.
numMaxArticoliPerRevisore	INTEGER	NOT NULL	Valore che delimita il valore massimo di revisori per articolo.
rangeValutazione	INTEGER	NOT NULL	Valore che delimita il valore per le valutazioni.
dataScadenzaInvioArticoli	DATETIME	NOT NULL	Delimita la data di scadenza per inviare gli articoli.
dataScadenzaInvioRevisioni	DATETIME	NOT NULL	Delimita la data di scadenza per inviare le revisioni.
dataScadenzaAccettazioneArticolo	DATETIME	NOT NULL	Delimita la data di scadenza per l'accettazione degli articoli.

dataScadenzaInvioVersioniFinali	DATETIME	NOT NULL	Delimita la data di scadenza per inviare le versioni finali degli articoli.
id_utente	INTEGER	FK, AUTO_INCREMENT	Identificativo univoco dell'utente.(Utente)
Id_Graduatoria	INTEGER	FK, AUTO_INCREMENT	Identificativo univoco per ogni voce di graduatoria. (Graduatoria)
Id_Assegnazione	INTEGER	FK, AUTO_INCREMENT	Identificativo univoco per ogni voce da assegnare. (AssegnazioneAutomatica)

Tabella: AssegnazioniAutomatiche (articolo,evento 1,1 1,1 avviene)

Nome Colonna	Tipo Dati	Vincoli	Descrizione
id_Assegnazione	INTEGER	PK, AUTO_INCREMENT	Identificativo univoco della assegnazione.
nomeArticolo	VARCHAR(50)	NOT NULL	Nome dell'articolo
nomeRevisore	VARCHAR(50)	NOT NULL	Nome del revisore assegnato.
cognomeRevisore	VARCHAR(50)	NOT NULL	Cognome del revisore assegnato.
nomeEvento	VARCHAR(255)	NOT NULL	Nome dell'evento.
titoloAssegnazioneEvento	VARCHAR(50)	NOT NULL	Titolo dell'assegnazione in base all'evento assegnato.
id_evento	INTEGER	FK, AUTO_INCREMENT	Chiave primaria univoca per ogni evento.
id_articolo	INTEGER	NOT NULL, FK	Identificativo dell'articolo a cui si riferisce questa voce di graduatoria.

id_utente	INTEGER	FK, AUTO_INCREMENT	Identificativo univoco dell'utente.(Utente)
------------------	---------	-----------------------	--