

PROGETTO BASI DI DATI 2021/2022

Francesco Guerrini 0000924798

Federica Mistura 0000922595

Stefano Monti 0000948005

1) RACCOLTA/ANALISI DEI REQUISITI

● Specifica

La piattaforma CONFVIRTUAL deve gestire i dati delle conferenze a tema informatica/tecnologie digitali.

Ogni conferenza dispone di un nome (es. "International Conference on Database Systems"), un acronimo (es. "DB"), un anno di edizione, delle date di svolgimento (uno o più giorni), un'immagine per il logo, un campo svolgimento (campo enum con due possibili valori: "Attiva" o "Completata"), un campo #totale_sponsorizzazioni. Anno di edizione ed acronimo (es. "DB 2021") sono univoci nella piattaforma. La conferenza può disporre di uno o più sponsor; ogni sponsor dispone di un nome, un'immagine per il logo, ed eroga un importo per la sponsorizzazione di ogni convegno in cui appare. Lo stesso sponsor può apparire in più conferenze. (Uno sponsor eroga lo stesso importo per ogni convegno).

Per ogni giorno di svolgimento della conferenza, si vuole tenere traccia del programma giornaliero. Quest'ultimo è strutturato in una o più sessioni.

Ogni sessione ha un codice (univoco, a livello globale), un titolo (stringa di 100 caratteri), un campo #numero_presentazioni, un'ora di inizio un'ora di fine, ed un link alla stanza Teams per lo svolgimento; ogni sessione è allocata ad un solo programma giornaliero. Invece, è consentito avere sessioni in parallelo (stessa data ed orario inizio/fine sovrapposte).

La sessione è composta da una o più presentazioni: ogni presentazione dispone di un codice (univoco), un'ora di inizio, un'ora di fine ed un numero di sequenza all'interno della sessione; non è possibile avere presentazioni che eccedano l'orario di inizio o di fine della propria sessione. Le presentazioni possono appartenere a due tipologie, in maniera esclusiva: presentazioni di articoli o tutorial. Nel primo caso, si vuole tenere traccia del titolo dell'articolo, del numero di pagine e del file PDF, della lista degli autori, e della lista di parole-chiave; ogni parola chiave è una stringa di massimo 20 caratteri, ogni autore dispone di nome e cognome. Inoltre, ogni presentazione di articoli dispone di campo stato_svolgimento (campo enum con due possibili valori: "Coperto" o "Non coperto"). Nel caso di tutorial, si vuole tenere traccia del titolo dell'intervento e dell'abstract (campo testuale di 500 caratteri).

La piattaforma consente iscrizioni volontarie da parte degli utenti; ogni utente dispone di username, password, nome, cognome, data e luogo di nascita. Sono previste tre tipologie di utenti: presenter, speaker ed amministratore. Esistono tuttavia utenti generici che non appartengono ad alcuna delle tre categorie sopra menzionate. Ogni utente può registrarsi alle conferenze presenti nella piattaforma; lo stesso utente può essere registrato ad un numero arbitrario di conferenze, una conferenza può disporre di un numero arbitrario di utenti registrati.

Sia i presenter sia gli speaker dispongono di un curriculum vitae (campo testo di 30 caratteri), una foto e un'affiliazione universitaria, con nome dell'università e nome del dipartimento. I presenter possono svolgere SOLO presentazioni di articoli; ogni presentazione di articolo è associata ad uno ed un solo presenter. Un presenter può presentare più articoli. Un presenter deve essere necessariamente uno degli autori dell'articolo. Gli speaker possono svolgere SOLO presentazioni di tutorial; ogni presentazione di tutorial è associata ad ALMENO uno speaker. Uno speaker può presentare più tutorial. Inoltre, è prevista la possibilità per uno speaker (e SOLO a loro) di caricare sulla piattaforma delle risorse aggiuntive per consentire la fruizione del proprio tutorial: una risorsa aggiuntiva consiste in un link Web (es. link a video Youtube), in una breve descrizione di testo e fa riferimento ad un solo tutorial. Un tutorial potrebbe non avere risorse aggiuntive. Gli amministratori (e solo loro) possono creare le conferenze presenti nella piattaforma; ogni conferenza è associata ad ALMENO un amministratore. Un amministratore può creare più conferenze. All'atto della creazione di una nuova conferenza, l'amministratore è anche automaticamente registrato alla stessa (vedi sopra). Gli utenti amministratori (e solo loro) possono inserire una valutazione sulle presentazioni: ogni valutazione dispone di voto (intero tra 0 e 10) e note (stringa con massimo 50 caratteri). Infine, ogni utente (di qualsiasi categoria) dispone di una propria lista di "presentazioni favorite", ossia un reminder delle presentazioni del convegno che si intende seguire; la lista può includere zero, uno o più elementi. Ogni sessione del convegno dispone di una chat, con possibilità di invio di messaggi da parte degli utenti (di qualsiasi categoria); ogni messaggio dispone di una data di inserimento (Timestamp messaggio) ed un testo. Uno stesso utente può inserire più di un messaggio nella stessa chat. La chat consente l'inserimento di messaggi solo nell'orario di inizio della sessione, e si disattiva immediatamente dopo l'orario di fine della stessa.

● **Lista delle operazioni**

Operazioni che riguardano tutti gli UTENTI:

- Autenticazione alla piattaforma
- Registrazione alla piattaforma
- Visualizzazione delle conferenze disponibili
- Registrazione ad una conferenza
- Visualizzazione programmi giornaliere delle conferenze
- Visualizzazione delle sessioni e delle presentazioni in ogni sessione
- Visualizzazione parole chiave e autori relativi a ogni presentazione di tipo Articolo
- Visualizzazione speaker e risorse aggiuntive relativi a ogni presentazione di tipo Tutorial
- Visualizzazione/inserimento messaggi nella chat di sessione
- Inserimento/visualizzazione lista presentazioni favorite delle conferenze a cui sei registrato
- Visualizzazione/modifica informazioni personali

Operazioni che riguardano SOLO gli utenti AMMINISTRATORE:

(Tutte le operazioni riguardano solo le conferenze che un amministratore può gestire ovvero quelle che ha creato lui oppure a quelle a cui è stato aggiunto come collaboratore)

- Creazione di una nuova conferenza
- Eliminazione di una conferenza
- Aggiungere una nuova data relativa a una conferenza
- Creazione di una nuova sessione della conferenza
- Eliminazione di una sessione
- Inserimento delle presentazioni in una sessione
- Eliminazione di una presentazione
- Aggiungere un nuovo amministratore per una delle proprie conferenze
- Associazione di uno speaker alla presentazione di un tutorial
- Associazione di un presenter alla presentazione di un articolo
- Associazione autore articolo
- Inserimento/visualizzazione delle valutazioni
- Inserimento di uno sponsor/sponsorizzazione
- Inserimento parole chiave per un articolo

Operazioni che riguardano SOLO PRESENTER:

- Inserimento/modifica del CV/foto ed affiliazione universitaria
- Visualizzazione Articoli da presentare

Operazioni che riguardano SOLO gli UTENTI SPEAKER:

- Inserimento/modifica del CV/foto ed affiliazione universitaria
- Inserimento/modifica delle risorse aggiuntive per il proprio tutorial
- Visualizzazione Tutorial da presentare

Statistiche (visibili da tutti gli utenti nella pagina iniziale):

- Visualizzare il numero (totale) di conferenze registrate
- Visualizzare il numero (totale) di conferenze attive
- Visualizzare il numero (totale) di utenti registrati
- Visualizzare la classifica dei presenter/speaker sulla base del voto medio

● **Tavola media dei volumi**

- Inserire una nuova sponsorizzazione per una data conferenza (2 volte/mese, interattiva)
 - Rimuovere una conferenza e tutte le proprie sponsorizzazioni (1 volta/mese, batch)
 - Calcolare e visualizzare il totale delle sponsorizzazioni di una data conferenza (4 volte/mese, interattiva)
 - Tabella dei volumi: 10 conferenze, 5 sponsor per conferenza.
- α (peso operazioni scrittura) =2, w_I (peso operazioni interattive) =1, w_B (peso operazioni batch) =0.5

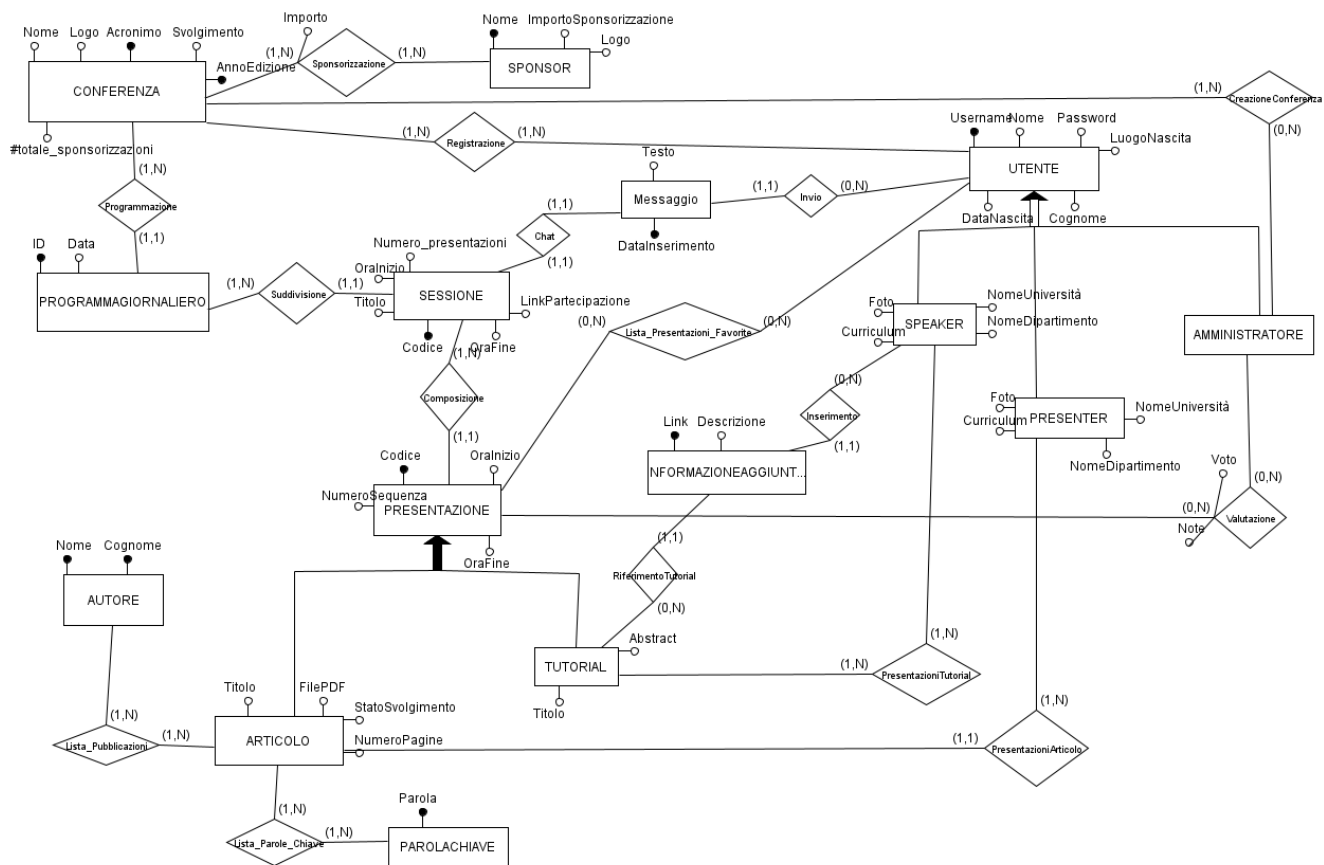
● **Glossario dei termini**

Termine	Descrizione	Sinonimo	Collegamento
Conferenza	Discussione con argomenti a tema informatico/tecnologico	Convegno	Sponsor, Programma Giornaliero, Utente, Amministratore
Sponsor	Società che sponsorizza una conferenza		Conferenza
Programma Giornaliero	Eventi di un determinato giorno		Conferenza, Sessione
Messaggio	Scambio di informazioni/opinioni tra utenti di una stessa sessione		Sessione, Utente
Sessione	Sessioni in cui è divisa la conferenza in una determinata data		Programma giornaliero, Presentazione
Utente	Persona che partecipa a una conferenza	Spettatore	Conferenza, Messaggio, Presentazione, Amministratore, Speaker, Presenter
Amministratore	Tipologia di utente che gestisce determinate Conferenze		Utente, Conferenza, Valutazione
Speaker	Tipologia di Utente che tiene le presentazioni dei Tutorial		Utente, Tutorial, Risorsa Aggiuntiva
Presenter	Tipologia di Utente che tiene le presentazioni degli Articoli		Utente, Articolo
Valutazione	Voto attribuito a una presentazione	Parere	Amministratore
Risorsa Aggiuntiva	Risorsa che riguarda alcuni tutorial		Tutorial
Autore	Redattore dell'articolo		Articolo
Parola chiave	Parole che identificano al meglio un articolo		Articolo

Presentazione	Discussione di un determinato argomento all'interno di una determinata sessione	Esposizione	Sessione, Tutorial, Articolo
Tutorial	Tipologia di presentazione		Presentazione
Articolo	Tipologia di presentazione		Presentazione

2) PROGETTAZIONE CONCETTUALE

● Diagramma E-R



● Dizionario Entità

ENTITA'	DESCRIZIONE	ATTRIBUTI	IDENTIFICATORI
---------	-------------	-----------	----------------

CONFERENZA	Conferenza a tema informatico/Tecnologico digitale	Nome Acronimo Svolgimento AnnoEdizione DataFine DataInizio Logo TotaleSponsorizzazioni	Acronimo AnnoEdizione
SPONSOR	Insieme delle aziende che sponsorizzano una conferenza	Nome Logo ImportoSponsorizzazione	Nome
PROGRAMMAGIORNALIERO	Tiene traccia dei giorni programmati per le conferenze	ID Data	ID
SESSIONE	Sessioni da cui è composto il programma giornaliero	NumeroPresentazioni Orainizio OraFine LinkPartecipazione Codice Titolo	Codice
MESSAGGIO	Messaggio inserito dall'utente nella chat	DataInserimento Testo	DataInserime nto
UTENTE	Utenti che partecipano/organizzano le conferenze	Username Nome Password LuogoNascita Cognome DataNascita	Username
SPEAKER	Generalizzazione di UTENTI	Foto Curriculum NomeUniversità	-
PRESENTER	Generalizzazione di UTENTI	Foto Curriculum NomeDipartimento NomeUniversità	-

AMMINISTRATORE	Generalizzazione di UTENTI	-	-
PRESENTAZIONE	Presentazioni tenute per ogni sessione	Codice NumeroSequenza OraInizio OraFine	Codice
TUTORIAL	Generalizzazione di PRESENTAZIONE	Abstract Titolo	-
ARTICOLI	Generalizzazione di PRESENTAZIONE	NumeroPagine StatoSvolgimento TitoloArticolo FilePDF	-
AUTORE	Lista degli autori	Nome Cognome	Nome Cognome
INFORMAZIONIAGGIUNTIVE	Informazioni su un dato tutorial	Link Descrizione	Link

- **Dizionario relazioni**

RELAZIONE	DESCRIZIONE	CAMPI	ATTRIBUTI
SPONSORIZZAZIONE	Ogni conferenza dispone di uno o più sponsor	CONFERENZA SPONSOR	Importo
REGISTRAZIONE	Ogni conferenza ha degli utenti registrati	CONFERENZA UTENTE	-

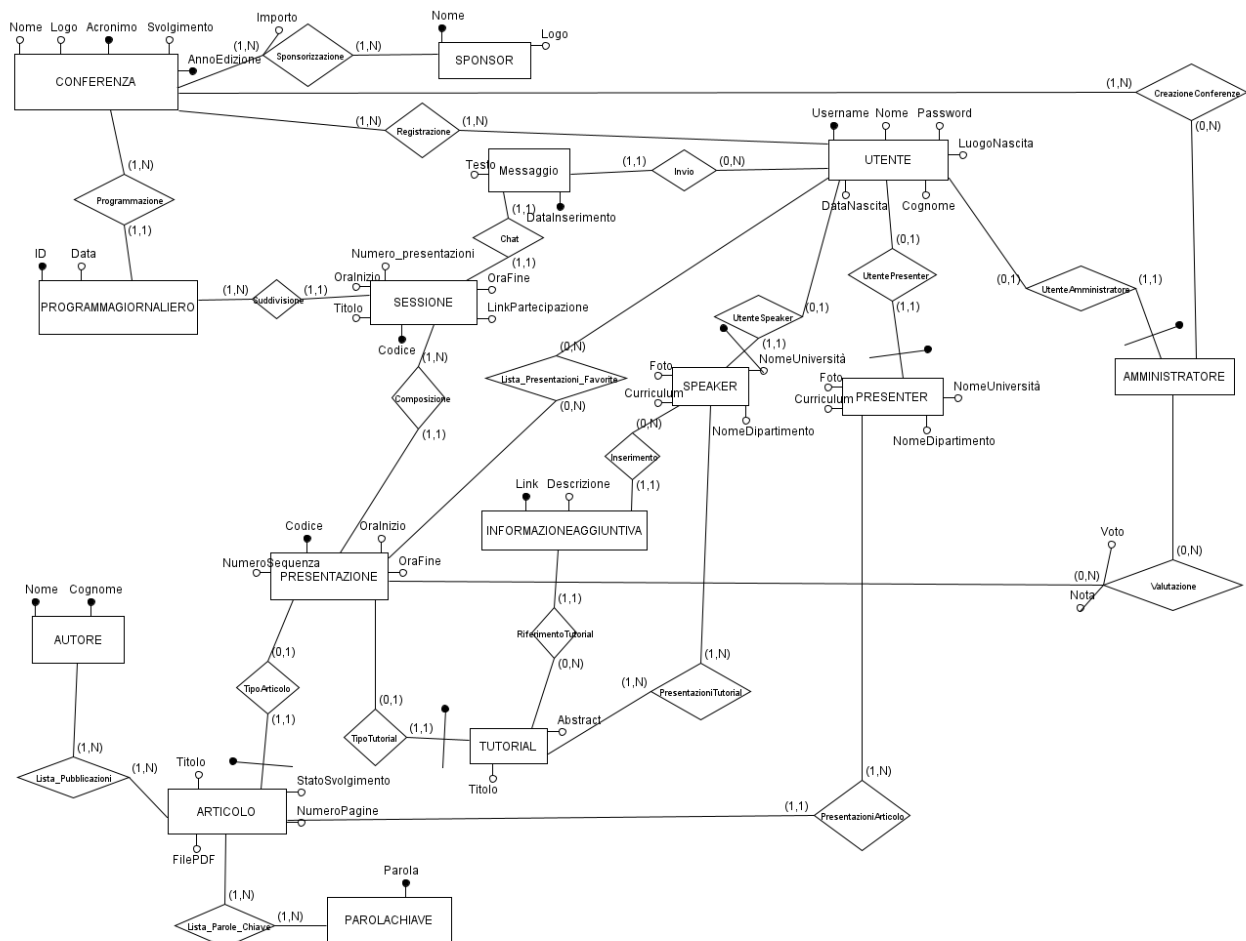
CREAZIONE_CONFERENZA	Ogni conferenza è creata da un amministratore	CONFERENZA AMMINISTRATORE	-
PROGRAMMAZIONE	Ogni conferenza ha il proprio programma	CONFERENZA PROGRAMMAGIORNALIERO	-
SUDDIVISIONE	Ogni programma giornaliero ha delle sessioni	PROGRAMMAGIORNALIERO SESSIONE	-
COMPOSIZIONE	Ogni sessione può avere una o più presentazioni	SESSIONE PRESENTAZIONE	-
VALUTAZIONE	Valutazioni delle presentazioni	AMMINISTRATORE PRESENTAZIONE	Note Voto
PRESENTAZIONEARTICOLO	Presentazioni degli articoli	PRESENTER ARTICOLO	-
PRESENTAZIONETUTORIAL	Presentazioni dei tutorial	SPEAKER TUTORIAL	-
INSERIMENTO	Inserimenti delle informazioni aggiuntive	SPEAKER INFORMAZIONEAGGIUNTIVA	-
RIFERIMENTOTUTORIAL	Ogni informazione aggiuntiva fa riferimento ad un tutorial	INFORMAZIONEAGGIUNTIVA TUTORIAL	-
LISTA_PUBBLICAZIONI	Lista degli articoli pubblicati da ogni autore	AUTORE ARTICOLO	-
CHAT	Ogni sessione ha una propria chat	SESSIONE MESSAGGIO	-
INVIO	Invio del messaggio	UTENTE MESSAGGIO	-
LISTA_PAROLE_CHIAVE	Lista di tutte le parole chiave per ogni articolo	ARTICOLO PAROLA-CHIAVE	-
LISTA_PRESENTAZIONE_FAVORITE	Lista delle presentazioni favorite da ogni utente	UTENTE PRESENTAZIONE	-

- **Tabella delle Business Rules**

L'attributo Svolgimento dell'entità CONFERENZA è un enum con due possibili valori "Attiva" o "Completata".
L'attributo Titolo dell'entità SESSIONE è una stringa di al massimo 100 caratteri.
Non è possibile avere presentazioni che eccedano l'orario di inizio o di fine della propria sessione.
Ogni parola chiave associata ad un articolo è una stringa di massimo 20 caratteri.
La generalizzazione ARTICOLO dell'entità PRESENTAZIONI ha l'attributo StatoSvolgimento che è un tipo ENUM con due possibili valori "Coperto" o "NonCoperto".
La generalizzazione TUTORIAL dell'entità PRESENTAZIONI ha un attributo chiamato Abstract che è un campo testuale con al massimo 500 caratteri.
Non possono esserci presentazioni in parallelo per la stessa sessione.
Le generalizzazioni di UTENTI Speaker e Presenter dispongono dell'attributo CurriculumVitae che è un campo testuale con al massimo 30 caratteri.
Nella relazione Valutazione troviamo l'attributo Voto che deve essere un intero compreso tra 0 e 10.
Nella relazione Valutazione troviamo l'attributo Note che deve essere una stringa con al massimo 50 caratteri.
L'inserimento di messaggi è consentito solo nell'orario di inizio della sessione, e si disattiva immediatamente dopo l'orario di fine della stessa.
È consentito avere sessioni che si svolgono in parallelo.
Un presenter deve essere necessariamente uno degli autori dell'articolo.
All'atto della creazione di una nuova conferenza, l'amministratore è anche automaticamente registrato alla conferenza.

3) PROGETTAZIONE LOGICA

● Ristrutturazione diagramma E-R



CONFERENZA(AnnoEdizione,Acronimo,Nome,Logo,Svolgimento)

SPONSOR(Nome,Logo)

SPONSORIZZAZIONE(AnnoEdizioneConferenza,AcronimoConferenza,NomeSponsor,Importo)

PROGRAMMAGIORNALIERO(ID,Data,AnnoEdizioneConferenza,AcronimoConferenza)

SESSIONE(Codice,Titolo,LinkPartecipazione,OraInizio,OraFine,
NumeroPresentazioni,IdProgramma)

PRESENTAZIONE(Codice,NumeroSequenza,CodiceSessione,OraInizio,OraFine)

ARTICOLO(CodiceArticolo,Titolo,StatoSvolgimento,NumeroPagine,FilePDF,UsernamePresenter)

TUTORIAL(CodiceTutorial,Titolo,Abstract)

PRESENTAZIONE_TUTORIAL(CodiceTutorial,UsernameSpeaker)

AUTORE(Nome,Cognome)

PAROLACHIAVE(Parola)

LISTA_PUBBLICAZIONI(NomeAutore,CognomeAutore,CodiceArticolo)

LISTA_PAROLE_CHIAVE(CodiceArticolo,ParolaChiave)
 UTENTE(Username,Nome,Cognome>Password,LuogoNascita,DataNascita)
 AMMINISTRATORE(UsernameAmministratore)
 PRESENTER(UsernamePresenter,Foto,Curriculum,NomeDipartimento,NomeUniversità)
 SPEAKER(UsernameSpeaker,Foto,Curriculum,NomeDipartimento,NomeUniversità)
 CREAZIONECONFERENZE(UsernameAmministratore,AnnoEdizConf,AcronimoConf)
 REGISTRAZIONE(UsernameUtente,AcronimoConf,AnnoEdiziConf)
 LISTA_PRESENTAZIONI_PREFERITE(CodicePresentazione,UsernameUtente)
 VALUTAZIONE(UsernameAmministratore,CodicePresentazione,Voto,Nota)
 INFORMAZIONEAGGIUNTIVA(Link,Descrizione,CodiceTutorial,UsernameSpeaker)
 MESSAGGIO(DataInserimento,UsernameUtente,Testo,CodiceSessione)

Vincoli integrità referenziale

PROGRAMMAGIORNALIERO.AnnoEdizioneConferenza → CONFERENZA.AnnoEdizione
 PROGRAMMAGIORNALIERO.AcronimoConferenza → CONFERENZA.Acronimo
 SPONSORIZZAZIONE.AnnoEdizioneConferenza → CONFERENZA.AnnoEdizione
 SPONSORIZZAZIONE.AcronimoConferenza → CONFERENZA.Acronimo
 SPONSORIZZAZIONE.NomeSponsor → SPONSOR.Nome
 SPONSORIZZAZIONE.LogoSponsor → SPONSOR.Logo
 SESSIONE.IdProgramma → PROGRAMMAGIORNALIERO.ID
 PRESENTAZIONE.CodiceSessione → SESSIONE.Codice
 ARTICOLO.CodiceArticolo → PRESENTAZIONE.Codice
 ARTICOLO.UsernamePresenter → PRESENTER.UsernamePresenter
 TUTORIAL.CodiceTutorial → PRESENTAZIONE.Codice
 PRESENTAZIONI_TUTORIAL.CodiceTutorial → TUTORIAL.CodiceTutorial
 PRESENTAZIONI_TUTORIAL.UsernameSpeaker → SPEAKER.UsernameSpeaker
 LISTA_PUBBLICAZIONI.NomeAutore → AUTORE.Nome
 LISTA_PUBBLICAZIONI.CognomeAutore → AUTORE.Cognome
 LISTA_PUBBLICAZIONI.CodiceArticolo → ARTICOLO.CodiceArticolo
 LISTA_PAROLE_CHIAVE.CodiceArticolo → ARTICOLO.CodiceArticolo
 LISTA_PAROLE_CHIAVE.ParolaChiave → PAROLACHIAVE.Parola
 AMMINISTRATORE.UsernameAmministratore → UTENTE.Username
 PRESENTER.UsernamePresenter → UTENTE.Username
 SPEAKER.UsernameSpeaker → UTENTE.Username
 CREAZIONECONFERENZE.UsernameAmministratore →
 AMMINISTRATORE.UsernameAmministratore
 CREAZIONECONFERENZE.AnnoEdizConf → CONFERENZA.AnnoEdizione
 CREAZIONECONFERENZE.AcronimoConf → CONFERENZA.Acronimo
 VALUTAZIONE.CodicePresentazione → PRESENTAZIONE.Codice
 VALUTAZIONE.UsernameAmministratore → AMMINISTRATORE.UsernameAmministratore
 INFORMAZIONEAGGIUNTIVA.CodiceTutorial → TUTORIAL.CodiceTutorial
 INFORMAZIONEAGGIUNTIVA.UsernameSpeaker → SPEAKER.UsernameSpeaker
 MESSAGGIO.CodiceSessione → SESSIONE.Codice

MESSAGGIO.UsernameUtente → UTENTE.Username

- **Analisi delle ridondanze**

Ridondanza = #totale_sponsorizzazioni

Tabella Operazioni

Operazione	Tipo	Frequenza
1) Inserire una nuova sponsorizzazione per una data conferenza	I	2 volte/mese
2) Rimuovere una conferenza e tutte le proprie sponsorizzazioni	B	1 volte/mese
3) Calcolare e visualizzare il totale delle sponsorizzazioni di una data conferenza	I	4 volte/mese

Analisi dei costi delle operazioni

Costanti:

- a (peso operazioni scrittura) = 2

- wI (peso operazioni interattive) = 1

- wB (peso operazioni batch) = 0.5

Formula: $c(O) = f(O) \cdot w \cdot (\alpha \cdot NC_{write} + NC_{read})$

CON RIDONDANZA:

$$C(op1) = 2 \cdot 1 \cdot (2 \cdot 2) = 8$$

$$C(op2) = 1 \cdot 0.5 \cdot (2 \cdot (1 + 5 + 5)) = 11$$

$$C(op3) = 4 \cdot 1 \cdot (5) = 20$$

$$C(S) = 39$$

SENZA RIDONDANZA:

$$C(op1) = 2 \cdot 1 \cdot (2 \cdot 1) = 4$$

$$C(op2) = 1 \cdot 0.5 \cdot (2 \cdot (1 + 5)) = 6$$

$$C(op3) = 4 \cdot 1 \cdot (5) = 20$$

$$C(S) = 30$$

Speed up = $30/39 = 0.76$ → rallentamento delle prestazioni mantenendo la ridondanza.

La ridondanza #totale_sponsorizzazioni conviene eliminarla infatti in base alle operazioni da effettuare sul database comporta solo un rallentamento.

4) NORMALIZZAZIONE

La base di dati è stata costruita per non presentare ridondanze; quindi, riteniamo che la normalizzazione non debba essere realizzata.

5) DESCRIZIONE AD ALTO LIVELLO DELLE FUNZIONALITA' DELL'APPLICAZIONE

La piattaforma CONFVIRTUAL ha come obbiettivo la gestione dei dati relativi alle conferenze a tema informatico/tecnologico.

La piattaforma è stata progettata nella seguente maniera. Appena si avvia il programma, si apre la Homepage nella quale sono presenti una breve descrizione dell'applicazione e alcune statistiche relative alle Conferenze e agli Utenti. Inoltre, nella Homepage sono visualizzati un bottone per effettuare l'operazione di LOGIN alla piattaforma e un bottone REGISTRATI con associato un menu a tendina necessari per registrarsi alla piattaforma.

Spieghiamo meglio come avvengono le fasi di registrazione alla piattaforma e di login.

L'utente deve, innanzitutto, selezionare tramite il menu a tendina la tipologia di utente, ossia il ruolo che avrà in questa piattaforma. Può decidere fra quattro possibili alternative: Generico, Amministratore, Speaker, Presenter. Deve quindi premere il bottone registrati il quale lo reindirizza al form di registrazione, nel quale deve inserire i dati richiesti.

Il pulsante LOGIN permette ad un utente di accedere alla piattaforma, tramite le sue credenziali: username e password. Nel caso in cui una delle due voci sia errata, si genera un messaggio d'errore. Al contrario, se i dati inseriti sono corretti l'utente è indirizzato alla propria Dashboard.

Una volta effettuato l'accesso, ogni utente può compiere una serie di operazioni a seconda della sua tipologia.

Gli utenti di tipo Generico possono:

- Autenticarsi alla piattaforma
- Registrarsi alla piattaforma
- Visualizzare delle conferenze disponibili
- Registrarsi ad una conferenza
- Visualizzare programmi giornalieri delle conferenze
- Visualizzare delle sessioni e delle presentazioni in ogni sessione
- Visualizzare parole chiave e autori relativi a ogni presentazione di tipo Articolo
- Visualizzare speaker e risorse aggiuntive relativi a ogni presentazione di tipo Tutorial
- Visualizzare/inserire messaggi nella chat di sessione
- Inserire/visualizzare lista presentazioni favorite delle conferenze a cui sei registrato
- Visualizzare/modificare informazioni personali

Tutte le operazioni di un utente Generico riguardano solo le conferenze a cui si è iscritto.

Gli utenti di tipo Amministratore possono:

- Svolgere tutte le operazioni di un utente generico;

- Gestire le Conferenze (Creazione di una Conferenza, eliminazione di una Conferenza, Inserimento di una nuova data per una Conferenza);
- Gestire le Sessioni (Eliminazione e creazione delle Sessioni);
- Gestire le Presentazioni (Eliminazione e creazione delle Presentazioni);
- Valutare le Presentazioni;
- Aggiungere un Amministratore per una delle proprie Conferenze;
- Associare uno Speaker e un Tutorial;
- Associare un Articolo ad un Presenter;
- Inserire/associare un Articolo a un Autore;
- Inserire/associare una Parola Chiave a un Articolo;
- Inserire un nuovo Sponsor;
- Inserire una nuova Sponsorizzazione.

Tutte le operazioni di un amministratore riguardano solo le conferenze che può gestire, ovvero quelle che ha creato lui oppure quelle a cui è stato aggiunto come collaboratore.

Gli utenti di tipo Presenter possono:

- Svolgere tutte le operazioni di un utente generico;
- Inserire/modificare CV/foto ed affiliazione universitaria;
- Visualizzare gli Articoli che devono presentare.

Gli utenti di tipo Speaker possono:

- Svolgere tutte le operazioni di un utente generico;
- Inserire/modificare CV/foto ed affiliazione universitaria;
- Inserire o aggiornare le Risorse Aggiuntive relative a un Tutorial;
- Visualizzare i Tutorial che devono presentare.

Ogni Utente è dotato di una propria sessione che viene salvata nel momento in cui avviene il Login e viene eliminata nel momento in cui si effettua il Log Out. La sessione consente di riconoscere il tipo di utente e di effettuare soltanto le operazioni che gli sono consentite, inoltre permette di reindirizzare l'utente alla propria dashboard.

Ogni dashboard è composta da una serie di bottoni ognuno dei quali è relativo a una specifica operazione. Quando l'utente clicca un bottone la piattaforma lo reindirizza nella pagina relativa alla operazione. Successivamente, una volta terminata l'operazione, l'utente può poi tornare alla propria Dashboard tramite l'apposito link. Una volta terminata la propria attività, l'utente può disconnettersi e tornare alla Homepage premendo il bottone di Log Out.

Tutti gli eventi presenti nella piattaforma, relativamente all'inserimento di nuovi dati sono memorizzati all'interno di un log, implementato in un'apposita collezione MongoDB. Il log è composto: dall'utente che ha effettuato l'operazione, dalla CALL che è stata chiamata e dal Timestamp relativo alla CALL.

Per rendere l'interfaccia gradevole, il front-end della nostra applicazione è stato realizzato tramite l'implementazione di CSS.

6) CODICE SQL

```
DROP DATABASE IF EXISTS CONFVIRTUAL;  
CREATE DATABASE CONFVIRTUAL;  
USE CONFVIRTUAL;
```

```
SET GLOBAL EVENT_SCHEDULER = ON;  
SET SQL_SAFE_UPDATES=0;  
SET @@global.event_scheduler =1;
```

```
/* -----*/  
/* Definizione delle tabelle SQL */
```

```
CREATE TABLE CONFERENZA(  
AnnoEdizione YEAR,  
    Acronimo VARCHAR(20),  
    Nome VARCHAR(30) NOT NULL,  
    Logo BLOB,  
    Svolgimento ENUM("ATTIVA" , "COMPLETATA") DEFAULT "ATTIVA",  
  
PRIMARY KEY(AnnoEdizione,Acronimo)  
) ENGINE = INNODB ;
```

```
CREATE TABLE SPONSOR(  
Nome VARCHAR(30) PRIMARY KEY,  
Logo BLOB  
  
) ENGINE = INNODB;
```

```
CREATE TABLE SPONSORIZZAZIONE(  
AnnoEdizioneConferenza YEAR,  
    AcronimoConferenza VARCHAR(20),  
    NomeSponsor VARCHAR(30) ,  
    Importo FLOAT NOT NULL,  
  
    PRIMARY KEY(AnnoEdizioneConferenza,AcronimoConferenza,NomeSponsor),  
    FOREIGN KEY(AnnoEdizioneConferenza,AcronimoConferenza) REFERENCES  
CONFERENZA(AnnoEdizione,Acronimo) on delete cascade,  
    FOREIGN KEY(NomeSponsor) REFERENCES SPONSOR(Nome) on delete cascade  
  
)ENGINE = INNODB;
```

```
CREATE TABLE PROGRAMMAGIORNALIERO(  
ID INT AUTO_INCREMENT ,  
Data DATE NOT NULL ,  
    AnnoEdizioneConferenza YEAR NOT NULL ,  
    AcronimoConferenza VARCHAR(20) NOT NULL,  
  
    PRIMARY KEY(ID),  
    UNIQUE KEY idUnico (Data,AnnoEdizioneConferenza,AcronimoConferenza) ,  
FOREIGN KEY(AnnoEdizioneConferenza,AcronimoConferenza) REFERENCES  
CONFERENZA(AnnoEdizione,Acronimo) on delete cascade  
  
    )ENGINE = INNODB;
```

```
CREATE TABLE SESSIONE(  
Codice CHAR(8) PRIMARY KEY,  
    Titolo VARCHAR(100),  
    LinkPartecipazione VARCHAR(100),  
    OraInizio TIME,  
    OraFine TIME,  
    IdProgramma INT,  
NumeroPresentazioni INT DEFAULT 0,  
    FOREIGN KEY(IdProgramma) REFERENCES PROGRAMMAGIORNALIERO(ID) on delete cascade  
    )ENGINE = INNODB;
```

```
CREATE TABLE UTENTE(  
Username VARCHAR(30) PRIMARY KEY,  
    Nome VARCHAR(30),  
    Cognome VARCHAR(30),  
    Password VARCHAR(12),  
    LuogoNascita VARCHAR(30),  
    DataNascita DATE  
    )ENGINE = INNODB;
```

```
CREATE TABLE AMMINISTRATORE(  
UsernameAmministratore VARCHAR(30) PRIMARY KEY,  
    FOREIGN KEY(UsernameAmministratore) REFERENCES UTENTE(Username)  
    )ENGINE = INNODB;
```



```
CREATE TABLE PRESENTER(  
UsernamePresenter VARCHAR(30) PRIMARY KEY,  
Foto BLOB,  
Curriculum VARCHAR(30),  
NomeDipartimento VARCHAR(30),  
NomeUniversita VARCHAR(40),  
FOREIGN KEY(UsernamePresenter) REFERENCES UTENTE(Username)  
)ENGINE = INNODB;
```

```
CREATE TABLE SPEAKER(  
UsernameSpeaker VARCHAR(30) PRIMARY KEY,  
Foto BLOB,  
Curriculum VARCHAR(30),  
NomeDipartimento VARCHAR(30),  
NomeUniversita VARCHAR(40),  
FOREIGN KEY (UsernameSpeaker) REFERENCES UTENTE(Username)  
)ENGINE = INNODB;
```

```
CREATE TABLE PRESENTAZIONE(  
Codice CHAR(7) PRIMARY KEY,  
NumeroSequenza INT,  
CodiceSessione CHAR(8),  
OraInizio TIME,  
OraFine TIME,  
  
FOREIGN KEY(CodiceSessione) REFERENCES SESSIONE(Codice) on delete cascade,  
UNIQUE KEY presentNonSovrapposta (CodiceSessione,NumeroSequenza)  
  
)ENGINE = INNODB;
```

```
CREATE TABLE ARTICOLO(  
CodiceArticolo CHAR(7) PRIMARY KEY ,  
Titolo VARCHAR(100),  
StatoSvolgimento ENUM("COPERTO" , "NON COPERTO") DEFAULT "NON COPERTO",  
FilePDF BLOB,  
NumeroPagine INT,  
UsernamePresenter VARCHAR(30),
```

```
FOREIGN KEY(CodiceArticolo) REFERENCES PRESENTAZIONE(Codice) on delete cascade,  
FOREIGN KEY(UsernamePresenter) REFERENCES PRESENTER(UsernamePresenter)  
) ENGINE = INNODB;
```

```
CREATE TABLE TUTORIAL(  
CodiceTutorial CHAR(7) PRIMARY KEY,  
Titolo VARCHAR(200),  
Abstract VARCHAR(500),
```

```
FOREIGN KEY(CodiceTutorial) REFERENCES PRESENTAZIONE(Codice) on delete cascade  
) ENGINE = INNODB;
```

```
CREATE TABLE PRESENTAZIONE_TUTORIAL(  
CodiceTutorial CHAR(7) ,  
UsernameSpeaker VARCHAR(30),  
  
PRIMARY KEY(CodiceTutorial,Usernamespeaker),  
FOREIGN KEY(CodiceTutorial) REFERENCES TUTORIAL(CodiceTutorial ) on delete cascade,  
FOREIGN KEY(UsernameSpeaker) REFERENCES SPEAKER(UsernameSpeaker)  
) ENGINE = INNODB;
```

```
CREATE TABLE AUTORE(  
Nome VARCHAR(30),  
Cognome VARCHAR(30),  
PRIMARY KEY(Nome, Cognome)  
)ENGINE = INNODB;
```

```
CREATE TABLE LISTA_PUBBLICAZIONI(  
NomeAutore VARCHAR(30),  
CognomeAutore VARCHAR(30),  
CodiceArticolo CHAR(7),  
PRIMARY KEY(NomeAutore, CognomeAutore, CodiceArticolo) ,  
FOREIGN KEY(NomeAutore,CognomeAutore) REFERENCES AUTORE(Nome,Cognome) on delete  
cascade ,  
FOREIGN KEY(CodiceArticolo) REFERENCES ARTICOLO(CodiceArticolo) on delete cascade  
)ENGINE = INNODB;
```

```
CREATE TABLE PAROLACHIAVE(  
Parola VARCHAR(20) PRIMARY KEY  
)ENGINE = INNODB;
```

```
CREATE TABLE LISTA_PAROLE_CHIAVE(  
CodiceArticolo CHAR(7),  
ParolaChiave VARCHAR(20),  
PRIMARY KEY(CodiceArticolo, ParolaChiave),  
FOREIGN KEY(CodiceArticolo) REFERENCES ARTICOLO(CodiceArticolo) on delete cascade,  
FOREIGN KEY(ParolaChiave) REFERENCES PAROLACHIAVE(Parola)  
)ENGINE = INNODB;
```

```
CREATE TABLE CREAZIONECONFERENZA(  
UsernameAmministratore VARCHAR(30),  
AnnoEdizConf YEAR,  
AcronimoConf VARCHAR(20),  
PRIMARY KEY(UsernameAmministratore,AnnoEdizConf,AcronimoConf),  
FOREIGN KEY(UsernameAmministratore) REFERENCES  
AMMINISTRATORE(UsernameAmministratore) ,  
FOREIGN KEY (AnnoEdizConf,AcronimoConf) REFERENCES  
CONFERENZA(AnnoEdizione,Acronimo) on delete cascade  
)ENGINE = INNODB;
```

```
CREATE TABLE REGISTRAZIONE(  
UsernameUtente VARCHAR(30),  
AcronimoConf VARCHAR(20),  
AnnoEdiziConf YEAR,  
PRIMARY KEY(UsernameUtente, AcronimoConf, AnnoEdiziConf),  
FOREIGN KEY(UsernameUtente) REFERENCES UTENTE(Username),  
FOREIGN KEY(AnnoEdiziConf,AcronimoConf) REFERENCES  
CONFERENZA(AnnoEdizione,Acronimo) on delete cascade  
)ENGINE = INNODB;
```

```
CREATE TABLE LISTA_PRESENTAZIONI_PREFERITE(  
CodicePresentazione CHAR(7),  
UsernameUtente VARCHAR(30),  
PRIMARY KEY(CodicePresentazione, UsernameUtente),  
FOREIGN KEY(CodicePresentazione) REFERENCES PRESENTAZIONE(Codice) on delete cascade,
```

```
FOREIGN KEY(UsernameUtente) REFERENCES UTENTE(Username)
)ENGINE = INNODB;
```

```
CREATE TABLE VALUTAZIONE(
UsernameAmministratore VARCHAR (30),
CodicePresentazione CHAR(7),
Voto INT CHECK(Voto>=0 and Voto<=10),
Note VARCHAR(50),
PRIMARY KEY(UsernameAmministratore, CodicePresentazione),
FOREIGN KEY (CodicePresentazione) REFERENCES PRESENTAZIONE(Codice) on delete cascade,
FOREIGN KEY (UsernameAmministratore) REFERENCES
AMMINISTRATORE(UsernameAmministratore)
)ENGINE = INNODB;
```

```
CREATE TABLE INFORMAZIONEAGGIUNTIVA(
Link VARCHAR(100) PRIMARY KEY,
Descrizione VARCHAR(255),
CodiceTutorial CHAR (7),
UsernameSpeaker VARCHAR(30),
FOREIGN KEY(CodiceTutorial) REFERENCES TUTORIAL(CodiceTutorial) on delete cascade,
FOREIGN KEY(UsernameSpeaker) REFERENCES SPEAKER(UsernameSpeaker)
)ENGINE = INNODB;
```

```
CREATE TABLE MESSAGGIO(
DataInserimento DATETIME,
UsernameUtente VARCHAR(30),
Testo VARCHAR(255),
CodiceSessione CHAR(8),
PRIMARY KEY(DataInserimento, UsernameUtente),
FOREIGN KEY(CodiceSessione) REFERENCES SESSIONE(Codice) on delete cascade,
FOREIGN KEY(UsernameUtente) REFERENCES UTENTE(Username)
)ENGINE = INNODB;
```

```
/*-----*/
```

```
#STORED PROCEDURE
```

```
#OPERAZIONI CHE RIGUARDANO TUTTI GLI UTENTI
```

```
#Registrare alla piattaforma un utente
```

```
START TRANSACTION
DELIMITER |
CREATE PROCEDURE registrazioneUtente(IN Username VARCHAR(30), Nome varchar(30),
Cognome VARCHAR(30),Password VARCHAR(12), LuogoNascita VARCHAR(30), DataNascita DATE)
BEGIN
INSERT INTO UTENTE() VALUES (Username, Nome, Cognome, Password, LuogoNascita,
DataNascita);
END |
DELIMITER ;
COMMIT;
```

#Registrare alla piattaforma un Amministratore

```
START TRANSACTION
DELIMITER |
CREATE PROCEDURE registrazioneAmministratore(IN UsernameAmministratore VARCHAR (30))
BEGIN
INSERT INTO AMMINISTRATORE() VALUES(UsernameAmministratore);
END |
DELIMITER ;
COMMIT;
```

#Registrare alla piattaforma uno Speaker

```
START TRANSACTION
DELIMITER |
CREATE PROCEDURE registrazioneSpeaker(IN UsernameSpeaker VARCHAR(30), Foto BLOB,
Curriculum VARCHAR(30), NomeDipartimento VARCHAR(30), NomeUniversita VARCHAR(40))
BEGIN
INSERT INTO SPEAKER() VALUES(UsernameSpeaker, Foto, Curriculum, NomeDipartimento,
NomeUniversita);
END |
DELIMITER ;
COMMIT;
```

#Registrare alla piattaforma un Presenter

```
START TRANSACTION
DELIMITER |
CREATE PROCEDURE registrazionePresenter(IN UsernamePresenter VARCHAR(30), Foto BLOB,
Curriculum VARCHAR(30), NomeDipartimento VARCHAR(30), NomeUniversita VARCHAR(40))
BEGIN
INSERT INTO PRESENTER() VALUES(UsernamePresenter, Foto, Curriculum, NomeDipartimento,
NomeUniversita);
END |
```

```
DELIMITER ;  
COMMIT;
```

```
#Registrazione ad una conferenza
```

```
START TRANSACTION
```

```
DELIMITER |
```

```
CREATE PROCEDURE registrazioneConferenza(IN UsernameUtente VARCHAR(30) , AcronimoConf  
VARCHAR(20), AnnoEdiziConf YEAR)
```

```
BEGIN
```

```
INSERT INTO REGISTRAZIONE() VALUES (UsernameUtente, AcronimoConf, AnnoEdiziConf);
```

```
END |
```

```
DELIMITER ;
```

```
COMMIT;
```

```
#Inserimento messaggi
```

```
START TRANSACTION
```

```
DELIMITER |
```

```
CREATE PROCEDURE inserimentoMessaggio(IN DataInserimento DATETIME, UsernameUtente  
VARCHAR(30), Testo VARCHAR(255), CodiceSessione CHAR(8))
```

```
BEGIN
```

```
INSERT INTO MESSAGGIO() VALUES(DataInserimento, UsernameUtente, Testo, CodiceSessione);
```

```
END |
```

```
DELIMITER ;
```

```
COMMIT;
```

```
#Inserimento lista presentazioni favorite
```

```
START TRANSACTION
```

```
DELIMITER |
```

```
CREATE PROCEDURE inserimentoPresentazioniFavorite(IN CodicePresentazione CHAR(7),  
UsernameUtente VARCHAR(30))
```

```
BEGIN
```

```
INSERT INTO LISTA_PRESENTAZIONI_PREFERITE() VALUES (CodicePresentazione,  
UsernameUtente);
```

```
END |
```

```
DELIMITER ;
```

```
COMMIT;
```

```
#Modifica info personali di un utente
```

```
START TRANSACTION
```

```
DELIMITER |
```

```
CREATE PROCEDURE modificaInfoPersonalì(IN ThisUsername VARCHAR(30) ,NewNome  
VARCHAR(30), NewCognome VARCHAR(30), NewPassword VARCHAR(30), NewLuogoNascita  
VARCHAR(40) , NewDataNascita VARCHAR(40))
```

```

BEGIN
UPDATE UTENTE
SET
Nome=NewNome,
    Cognome=NewCognome,
    Password=NewPassword,
    LuogoNascita=NewLuogoNascita,
    DataNascita=NewDataNascita
WHERE Username=ThisUsername;
END |
DELIMITER ;
COMMIT;

```

#OPERAZIONI CHE RIGUARDANO SOLO GLI UTENTI AMMINISTRATORI

#Creazione di una nuova conferenza + inserimento in CREAZIONECONFERENZA() + registrazione alla conferenza

```

START TRANSACTION
DELIMITER |
CREATE PROCEDURE creaConferenza(IN UsernameAmministratore VARCHAR(30), AnnoEdizConf
YEAR, AcronimoConf VARCHAR(20), Nome VARCHAR(30), Logo BLOB)
BEGIN
INSERT INTO CONFERENZA(AnnoEdizione,Acronimo,Nome,Logo) VALUES (AnnoEdizConf,
AcronimoConf,Nome,Logo);
INSERT INTO CREAZIONECONFERENZA() VALUES (UsernameAmministratore, AnnoEdizConf,
AcronimoConf);
END |
DELIMITER ;
COMMIT;

```

#Eliminazione di una conferenza

```

START TRANSACTION
DELIMITER |
CREATE PROCEDURE eliminaConferenza(IN AnnoEdizione YEAR, Acronimo VARCHAR(20))
BEGIN
DELETE FROM CONFERENZA WHERE(CONFERENZA.AnnoEdizione = AnnoEdizione AND
CONFERENZA.Acronimo = Acronimo);
IF (ROW_COUNT() = 0) then SIGNAL SQLSTATE '45000';
END IF;
END |
DELIMITER ;
COMMIT;

```

#Associazione conferenza amministratore

START TRANSACTION

DELIMITER |

CREATE PROCEDURE conferenzaAmministratoreAssoc(IN UsernameAmministratore VARCHAR(30),
AnnoEdizConf YEAR, AcronimoConf VARCHAR(20))

BEGIN

INSERT INTO CREAZIONECONFERENZA(UsernameAmministratore, AnnoEdizConf, AcronimoConf)
VALUES (UsernameAmministratore, AnnoEdizConf, AcronimoConf);

END |

DELIMITER ;

COMMIT;

#Procedura per l'inserimento di un PROGRAMMAGIORNALIERO nella tabella

START TRANSACTION

DELIMITER |

CREATE PROCEDURE inserisciProgrammaGiornaliero (IN Data DATE, AnnoEdizioneConferenza
YEAR, AcronimoConferenza VARCHAR(20))

BEGIN

INSERT INTO PROGRAMMAGIORNALIERO(Data, AnnoEdizioneConferenza, AcronimoConferenza)
VALUES (Data, AnnoEdizioneConferenza, AcronimoConferenza);

END |

DELIMITER ;

COMMIT;

#Creazione di una nuova sessione della conferenza

START TRANSACTION

DELIMITER |

CREATE PROCEDURE creazioneSessione(IN Codice CHAR(8), Titolo VARCHAR(100),
LinkPartecipazione VARCHAR(100),OraInizio TIME, OraFine TIME, IdProgramma INT)

BEGIN

INSERT INTO SESSIONE(Codice,Titolo,LinkPartecipazione,OraInizio,OraFine,IdProgramma)
VALUES(Codice, Titolo, LinkPartecipazione, OraInizio, OraFine, IdProgramma);

END |

DELIMITER ;

COMMIT;

#Eliminazione di una sessione

START TRANSACTION

DELIMITER |

CREATE PROCEDURE eliminaSessione(IN Codice CHAR(8))


```
BEGIN
DELETE FROM SESSIONE WHERE (SESSIONE.Codice = Codice);
IF (ROW_COUNT() = 0) then SIGNAL SQLSTATE '45000';
END IF;
END |
DELIMITER ;
COMMIT;
```

#Eliminazione di una presentazione

```
START TRANSACTION
DELIMITER |
CREATE PROCEDURE eliminaPresentazione(IN Codice CHAR(7))
BEGIN
DELETE FROM PRESENTAZIONE WHERE (PRESENTAZIONE.Codice = Codice);
IF (ROW_COUNT() = 0) then SIGNAL SQLSTATE '45000';
END IF;
END |
DELIMITER ;
COMMIT;
```

#Inserimento delle presentazioni in una sessione

```
START TRANSACTION
DELIMITER |
CREATE PROCEDURE inserimentoPresentazione(IN Codice CHAR(7), NumeroSequenza INT,
CodiceSessione CHAR(8), OraInizio TIME, OraFine TIME)
BEGIN
INSERT INTO PRESENTAZIONE() VALUES(Codice, NumeroSequenza, CodiceSessione, OraInizio,
OraFine);
END |
DELIMITER ;
COMMIT;
```

#Inserimento di un tutorial

```
START TRANSACTION
DELIMITER |
CREATE PROCEDURE inserimentoTutorial(IN CodiceTutorial CHAR(7), Titolo VARCHAR(200),
Abstract VARCHAR(500))
BEGIN
INSERT INTO TUTORIAL() VALUES (CodiceTutorial, Titolo, Abstract);
END |
DELIMITER ;
COMMIT;
```

#Associazione di uno speaker alla presentazione di un tutorial

START TRANSACTION

DELIMITER |

CREATE PROCEDURE speakerTutorialAssoc(IN CodiceTutorial CHAR(7), UsernameSpeaker VARCHAR(30))

BEGIN

INSERT INTO PRESENTAZIONE_TUTORIAL() VALUES (CodiceTutorial, UsernameSpeaker);

END |

DELIMITER ;

COMMIT;

#Inserimento articolo

START TRANSACTION

DELIMITER |

CREATE PROCEDURE inserimentoArticolo(IN CodiceArticolo CHAR(7), Titolo VARCHAR(100), FilePDF BLOB, NumeroPagine INT)

BEGIN

INSERT INTO ARTICOLO (CodiceArticolo, Titolo, FilePDF, NumeroPagine) VALUES (CodiceArticolo, Titolo, FilePDF, NumeroPagine);

END |

DELIMITER ;

COMMIT;

#Associazione di un presenter alla presentazione di un articolo

START TRANSACTION

DELIMITER |

CREATE PROCEDURE presenterArticoloAssoc(IN CodiceArticolo CHAR(7), UsernamePresenter VARCHAR(30))

BEGIN

UPDATE ARTICOLO SET ARTICOLO.UsernamePresenter = UsernamePresenter WHERE (ARTICOLO.CodiceArticolo = CodiceArticolo);

END |

DELIMITER ;

COMMIT;

#Inserimento delle valutazioni sulle presentazioni

START TRANSACTION

DELIMITER |

CREATE PROCEDURE inserisciValutazione(IN Note VARCHAR(50), Voto INT, UserAdmin VARCHAR(20), CodicePresentazione CHAR(7))

BEGIN

INSERT INTO VALUTAZIONE() VALUES (UserAdmin, CodicePresentazione,Voto, Note);

END |

```
DELIMITER ;  
COMMIT;
```

#Aggiunta di un amministratore per la gestione di una conferenza

```
START TRANSACTION  
DELIMITER |  
CREATE PROCEDURE aggiungiAmministratore(IN AcronimoConf VARCHAR(20), AnnoEdizioneConf  
YEAR, UsernameAmministratore VARCHAR(20))  
BEGIN  
INSERT INTO CREAZIONECONFERENZA() VALUES (UsernameAmministratore, AnnoEdizioneConf,  
AcronimoConf);  
END |  
DELIMITER ;  
COMMIT;
```

#Inserimento di uno sponsor

```
START TRANSACTION  
DELIMITER |  
CREATE PROCEDURE insertSponsor(IN Nome VARCHAR(30), Logo BLOB)  
BEGIN  
INSERT INTO SPONSOR() VALUES (Nome, Logo);  
END |  
DELIMITER ;  
COMMIT;
```

#Inserimento di una sponsorizzazione

```
START TRANSACTION  
DELIMITER |  
CREATE PROCEDURE inserimentoSponsorizzazione (IN AnnoEdizioneConferenza YEAR,  
AcronimoConferenza VARCHAR(20), NomeSponsor VARCHAR(30), Importo FLOAT)  
BEGIN  
INSERT INTO SPONSORIZZAZIONE() VALUES(AnnoEdizioneConferenza, AcronimoConferenza,  
NomeSponsor, Importo);  
END |  
DELIMITER ;  
COMMIT;
```

#Inserimento parola chiave

```
START TRANSACTION  
DELIMITER |  
CREATE PROCEDURE inserimentoParolaChiave(IN Parola VARCHAR(20))  
BEGIN
```

```
INSERT INTO PAROLACHIAVE() VALUES (Parola);  
END |  
DELIMITER ;  
COMMIT;
```

#Associazione di una parola chiave ad un articolo

```
START TRANSACTION  
DELIMITER |  
CREATE PROCEDURE articoloParolaChiaveAssoc(IN CodiceArticolo CHAR(7), ParolaChiave  
VARCHAR(20))  
BEGIN  
INSERT INTO LISTA_PAROLE_CHIAVE() VALUES(CodiceArticolo, ParolaChiave);  
END |  
DELIMITER ;  
COMMIT;
```

#Inserimento autore

```
START TRANSACTION  
DELIMITER |  
CREATE PROCEDURE inserimentoAutore(IN Nome VARCHAR(30), Cognome VARCHAR(30))  
BEGIN  
INSERT INTO AUTORE() VALUES (Nome, Cognome);  
END |  
DELIMITER ;  
COMMIT;
```

#Associazione di un autore ad un articolo

```
START TRANSACTION  
DELIMITER |  
CREATE PROCEDURE autoreArticoloAssoc(IN NomeAutore VARCHAR(30), CognomeAutore  
VARCHAR(30) , CodiceArticolo CHAR(7))  
BEGIN  
INSERT INTO LISTA_PUBBLICAZIONI() VALUES (NomeAutore, CognomeAutore , CodiceArticolo);  
END |  
DELIMITER ;  
COMMIT;
```

#OPERAZIONI CHE RIGUARDANO SOLO GLI UTENTI PRESENTER

#Modifica del CV/foto ed affiliazione universitaria

```
START TRANSACTION  
DELIMITER |
```

```

CREATE PROCEDURE modificaCVFotoUniPresenter(IN ThisUsernamePresenter VARCHAR(30),
NewPresenterFoto BLOB, NewPresenterCurriculum VARCHAR(30),
NewPresenterNomeDipartimento VARCHAR(30), NewPresenterNomeUniversita VARCHAR(40))
BEGIN
UPDATE PRESENTER
SET Foto=NewPresenterFoto,
UsernamePresenter=ThisUsernamePresenter,
Curriculum=NewPresenterCurriculum,
    NomeDipartimento=NewPresenterNomeDipartimento,
    NomeUniversita=NewPresenterNomeUniversita
WHERE ThisUsernamePresenter=UsernamePresenter;
END |
DELIMITER ;
COMMIT;

```

#Modifica del CV ed affiliazione universitaria

```

START TRANSACTION
DELIMITER |
CREATE PROCEDURE modificaCVUniPresenter(IN ThisUsernamePresenter VARCHAR(30),
NewPresenterCurriculum VARCHAR(30), NewPresenterNomeDipartimento VARCHAR(30),
NewPresenterNomeUniversita VARCHAR(40))
BEGIN
UPDATE PRESENTER
SET
UsernamePresenter=ThisUsernamePresenter,
Curriculum=NewPresenterCurriculum,
    NomeDipartimento=NewPresenterNomeDipartimento,
    NomeUniversita=NewPresenterNomeUniversita
WHERE ThisUsernamePresenter=UsernamePresenter;
END |
DELIMITER ;
COMMIT;

```

#OPERAZIONI CHE RIGUARDANO SOLO GLI UTENTI SPEAKER

#Modifica del CV/foto ed affiliazione universitaria

```

START TRANSACTION
DELIMITER |
CREATE PROCEDURE modificaCVFotoUniSpeaker(IN ThisUsernameSpeaker VARCHAR(30),
NewSpeakerFoto BLOB, NewSpeakerCurriculum VARCHAR(30), NewSpeakerNomeDipartimento
VARCHAR(30), NewSpeakerNomeUniversita VARCHAR(40))
BEGIN

```

```
UPDATE SPEAKER
SET Foto=NewSpeakerFoto,
Curriculum=NewSpeakerCurriculum,
    NomeDipartimento=NewSpeakerNomeDipartimento,
    NomeUniversita=NewSpeakerNomeUniversita
WHERE UsernameSpeaker = ThisUsernameSpeaker;
END |
DELIMITER ;
COMMIT;
```

#Modifica del CV ed affiliazione universitaria

```
START TRANSACTION
DELIMITER |
CREATE PROCEDURE modificaCVUniSpeaker(IN ThisUsernameSpeaker VARCHAR(30),
NewSpeakerCurriculum VARCHAR(30), NewSpeakerNomeDipartimento VARCHAR(30),
NewSpeakerNomeUniversita VARCHAR(40))
BEGIN
UPDATE SPEAKER
SET
Curriculum=NewSpeakerCurriculum,
    NomeDipartimento=NewSpeakerNomeDipartimento,
    NomeUniversita=NewSpeakerNomeUniversita
WHERE UsernameSpeaker = ThisUsernameSpeaker;
END |
DELIMITER ;
COMMIT;
```

#Inserimento/modifica delle risorse aggiuntive per il proprio tutorial

```
START TRANSACTION
DELIMITER |
CREATE PROCEDURE inserisciRisorsaAggiuntiva(IN Link VARCHAR(100), Descrizione
VARCHAR(255), CodiceTutorial CHAR(7), UsernameSpeaker VARCHAR(30))
BEGIN
INSERT INTO INFORMAZIONEAGGIUNTIVA() VALUES(Link, Descrizione, CodiceTutorial,
UsernameSpeaker);
END |
DELIMITER ;
COMMIT;
```

#Elimina risorse aggiuntive per il proprio tutorial

```
START TRANSACTION
```

```
DELIMITER |
CREATE PROCEDURE eliminaRisorsaAggiuntiva(IN ThisLink VARCHAR(100))
BEGIN
DELETE FROM INFORMAZIONEAGGIUNTIVA WHERE Link=ThisLink;
END |
DELIMITER ;
COMMIT;
```

```
#TRIGGER
#Utilizzare un trigger per implementare l'operazione cambio di stato_svolgimento di una
presentazione di articolo, portandolo
#da "non coperto" a "coperto" quando si inserisce un presenter valido per quella presentazione
START TRANSACTION
DELIMITER |
CREATE TRIGGER CambiamentoStatoSvolgimento
BEFORE UPDATE ON ARTICOLO
FOR EACH ROW
BEGIN
SET NEW.StatoSvolgimento = "COPERTO";
END |
DELIMITER ;
COMMIT;
```

```
#Utilizzare un trigger per implementare l'operazione di aggiornamento del campo
#numero_presentazioni ogni qualvolta si aggiunge
#una nuova presentazione ad una sessione della conferenza.
```

```
START TRANSACTION
DELIMITER |
CREATE TRIGGER AggiornaNumeroPresentazioni
AFTER INSERT ON PRESENTAZIONE
FOR EACH ROW
BEGIN
UPDATE SESSIONE
SET SESSIONE.NumeroPresentazioni = SESSIONE.NumeroPresentazioni + 1
WHERE (SESSIONE.Codice = NEW.CodiceSessione);
END |
DELIMITER ;
COMMIT;
```

#EVENTO

#Utilizzare un event per modificare il campo svolgimento di una conferenza. L'evento setta il campo a "Completata" non appena

#la data corrente eccede di un giorno l'ultima data di svolgimento di una conferenza

```
CREATE VIEW dataMax(AnnoEdizioneConferenza,AcronimoConferenza,DataUltima) AS (  
SELECT AnnoEdizioneConferenza, AcronimoConferenza, MAX(Data) AS DataUltima  
FROM PROGRAMMAGIORNALIERO, CONFERENZA  
WHERE (PROGRAMMAGIORNALIERO.AcronimoConferenza = CONFERENZA.Acronimo)  
AND (PROGRAMMAGIORNALIERO.AnnoEdizioneConferenza = CONFERENZA.AnnoEdizione)  
GROUP BY  
PROGRAMMAGIORNALIERO.AnnoEdizioneConferenza,PROGRAMMAGIORNALIERO.AcronimoConf  
erenza  
);  
SELECT * FROM dataMax;
```

DELIMITER \$

CREATE EVENT CambioStatoSvolgimento

ON SCHEDULE EVERY 1 MINUTE

DO

BEGIN

UPDATE CONFERENZA,dataMax

SET CONFERENZA.Svolgimento = "COMPLETATA"

WHERE(CONFERENZA.Acronimo = dataMax.AcronimoConferenza)

AND (CONFERENZA.AnnoEdizione = dataMax.AnnoEdizioneConferenza)

AND (dataMax.DataUltima < CURRENT_DATE());

END \$

DELIMITER ;