

# System Design Document

- Obiettivi del Sistema
- Architettura Software
- Package
  - Package conferecemanagementsystem
  - Package conferecemanagementsystem.Autore
  - Package conferecemanagementsystem.Chair
  - Package conferecemanagementsystem.Recensore
  - Package conferecemanagementsystem.Utils
- Database
  - Diagramma ER
  - Descrizione Tabelle
    - Descrizione Testuale
    - Tabella CONFERENZE
    - Tabella UTENTI
    - Tabella ARTICOLI
    - Tabella RECENSIONI
    - Tabella NOTIFICHE
    - Tabella AUTORI
    - Tabella COMITATO
- Accesso al Sistema

## Obiettivi del Sistema

CMS è un software di gestione delle Conferenze scientifiche, permette gestire tutti gli aspetti relativi alla conferenza: dalla creazione della stessa alla sottomissione degli articoli ed alla revisione degli stessi.

# Architettura Software

---

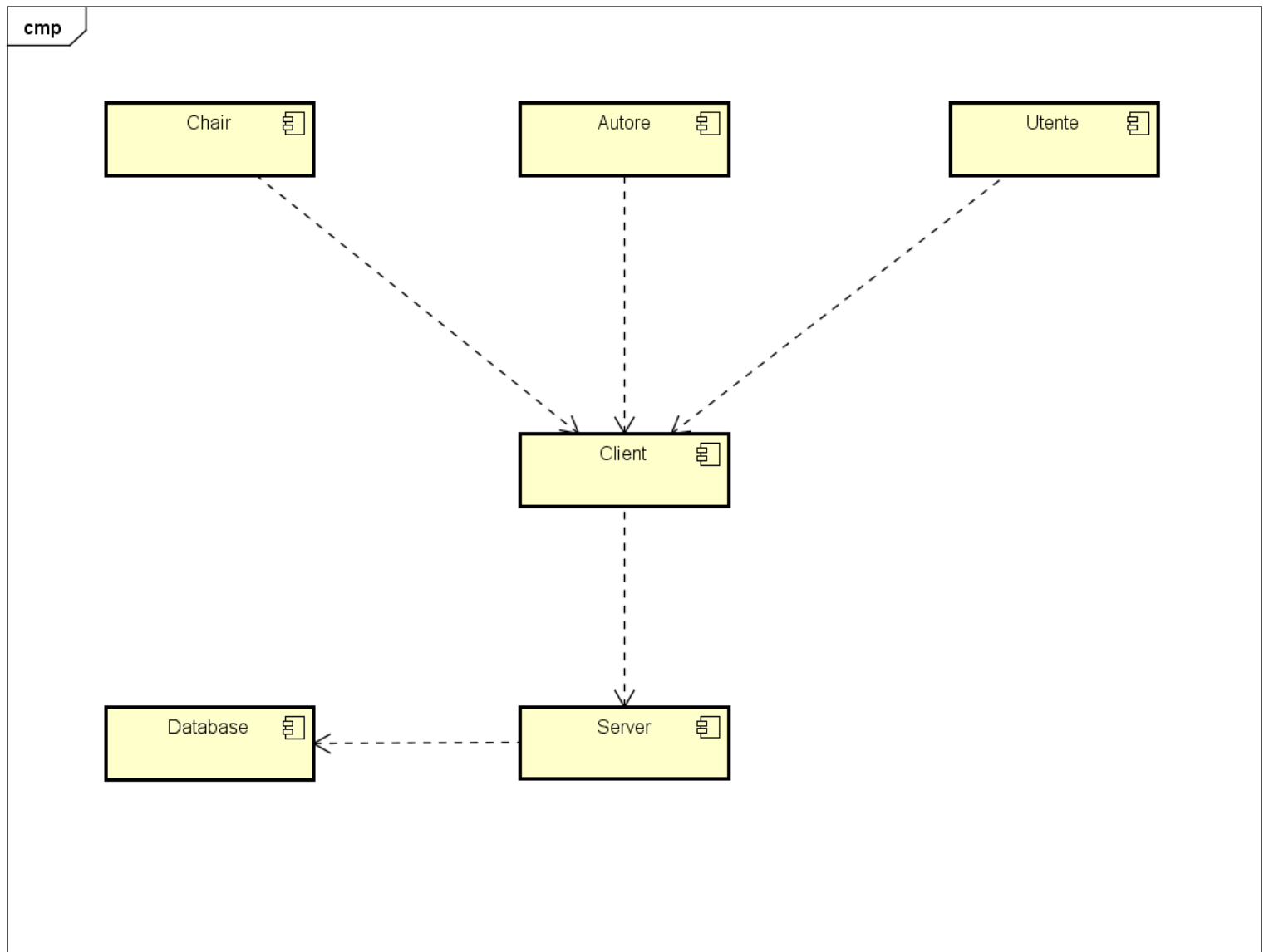
L'architettura proposta, in accordo con le specifiche, è di tipo Client-Server. Il lato Server avrà il compito di gestire il database, di tipo Relazionale implementato tramite la tecnologia MySQL. Il lato Client avrà il compito di interagire con gli utenti e di gestire le interazioni con il database. Il lato Client è implementato tramite le tecnologie JAVA, nello specifico si sono utilizzate le api SWING per la creazione delle GUI, le api java.time per la rappresentazione delle date, le api javax.mail (deprecata) per l'invio delle notifiche email, il driver JDBC per l'instaurazione della connessione col database MariaDB e le api jav.sql per le interrogazioni col database.

## Package

---

Il sistema proposto si compone di quattro package:

1. `conferecemanagementsystem.Autore`
2. `conferecemanagementsystem.Chair`
3. `conferecemanagementsystem.Recensore`
4. `conferecemanagementsystem.Utils`



## Package conferecemanagementsystem

Contiene una serie di Boundary e Control comuni a tutti gli utenti. In particolare contiene la boundary RegisterFrame che permette l'iscrizione alla conferenza.

Il package è così composto:

- \* CreaConferenzaFrame
- \* LoginFrame
- \* MainClass
- \* RegisterFrame
- \* StartFrame

La Control MainClass è responsabile dell'inizializzazione del sistema stesso. Essa si occupa di istanziare le entity conferenza, db, utente che verranno poi popolate dalle control implementate all'interno di LoginFrame.

La boundary CreaConferenzaFrame viene mostrata esclusivamente quando non esiste una conferenza all'interno del sistema: l'utente che crea fisicamente l'evento diviene (per il sistema) Chair di quella conferenza.

La boundary LoginFrame rappresenta la porta d'accesso al sistema. Al suo interno sono implementati i control che permettono al sistema di inizializzare coerentemente le entity conferenza, db, utente (create dalla control MainClass) e di determinare il ruolo dell'utente nella conferenza.

# Package

## conferecemanagementsystem.Autore

Contiene le Boundary che permettono all'autore di accedere alle funzionalità che CMS mette a disposizione degli autori.

Il package è così organizzato:

- \* AutoreFrame
  - \* Autore\_SottomettiFrame
  - \* Autore\_ListaArticoliFrame
  - \* Autore\_SottomettiRivistoFrame

La boundary AutoreFrame costituisce l'interfaccia iniziale con cui l'autore interagisce: tramite questa ha la possibilità di raggiungere tutte le funzionalità accessibili dall'autore. AutoreFrame è responsabile dei controlli temporali: l'autore può accedere a determinate funzioni solo in un prestabilito intervallo temporale.

# Package

## conferecemanagementsystem.Chair

Contiene le Boundary che permettono al Chair di accedere alle funzionalità che CMS mette a disposizione del chair.

Il package è così organizzato:

- \* ChairFrame
- \* Chair\_AutoriPartecipantiFrame
- \* Chair\_ComitatoDiProgrammaFrame
- \* Chair\_ComitatoDiProgramma\_AssegnaArticoliFrame
- \* Chair\_ComitatoDiProgramma\_InviaEmailFrame
- \* Chair\_ComitatoDiProgramma\_InvitaMembroFrame
- \* Chair\_ListaArticoliFrame
- \* Chair\_ListaRecensioniFrame
- \* Chair\_ListaRecensioni\_GraduatoriaFrame
- \* Chair\_NotificheDiSistemaFrame

La boundary ChairFrame costituisce l'interfaccia iniziale con cui il chair interagisce. Particolare importanza assumono le boundary Chair\_ListaRecensioni\_GraduatoriaFrame e Chair\_NotificheDiSistemaFrame: la prima implementa le Control che permettono di stilare la graduatoria finale degli articoli e di notificare gli esiti agli autori vincitori mentre la seconda costituisce la bountary con cui il chair viene tenuto aggiornato sugli eventi che si verificano nel sistema (sottomissione articoli, sottomissione recensioni, invito sottorecensori, rifiuto di una recensione).

## Package

# conferecemanagementsystem.Recensore

Contiene le Boundary che permettono al Recensore di accedere alle funzionalità che CMS mette a disposizione del recensore.

Il package è così organizzato:

- \* RecensoreFrame
- \* Recensore\_AccettaArticoliAssegnatiFrame
- \* Recensore\_InvitaSottoRecensoreFrame
- \* Recensore\_SottomettiRecensoreFrame

La boundary `RecensoreFrame` costituisce l'interfaccia iniziale con cui il recensore interagisce. `RecensoreFrame` implementa le `Control` che permettono al sistema di effettuare le verifiche temporali (è possibile sottomettere una recensione solo entro una certa data) e di accertare se il recensore sta partecipando alla conferenza anche come autore.

Di particolare importanza è la boundary `Recensore_InvitaSottoRecensoreFrame`. Tale boundary permette l'invito di un sottorecensore tuttavia il sottorecensore deve essere "accettato" dal chair: potrà sottomettere la recensione dell'articolo solo dopo che il chair glielo avrà assegnato.

## Package `conferecemanagementsystem.Utils`

Contiene le classi utilità.

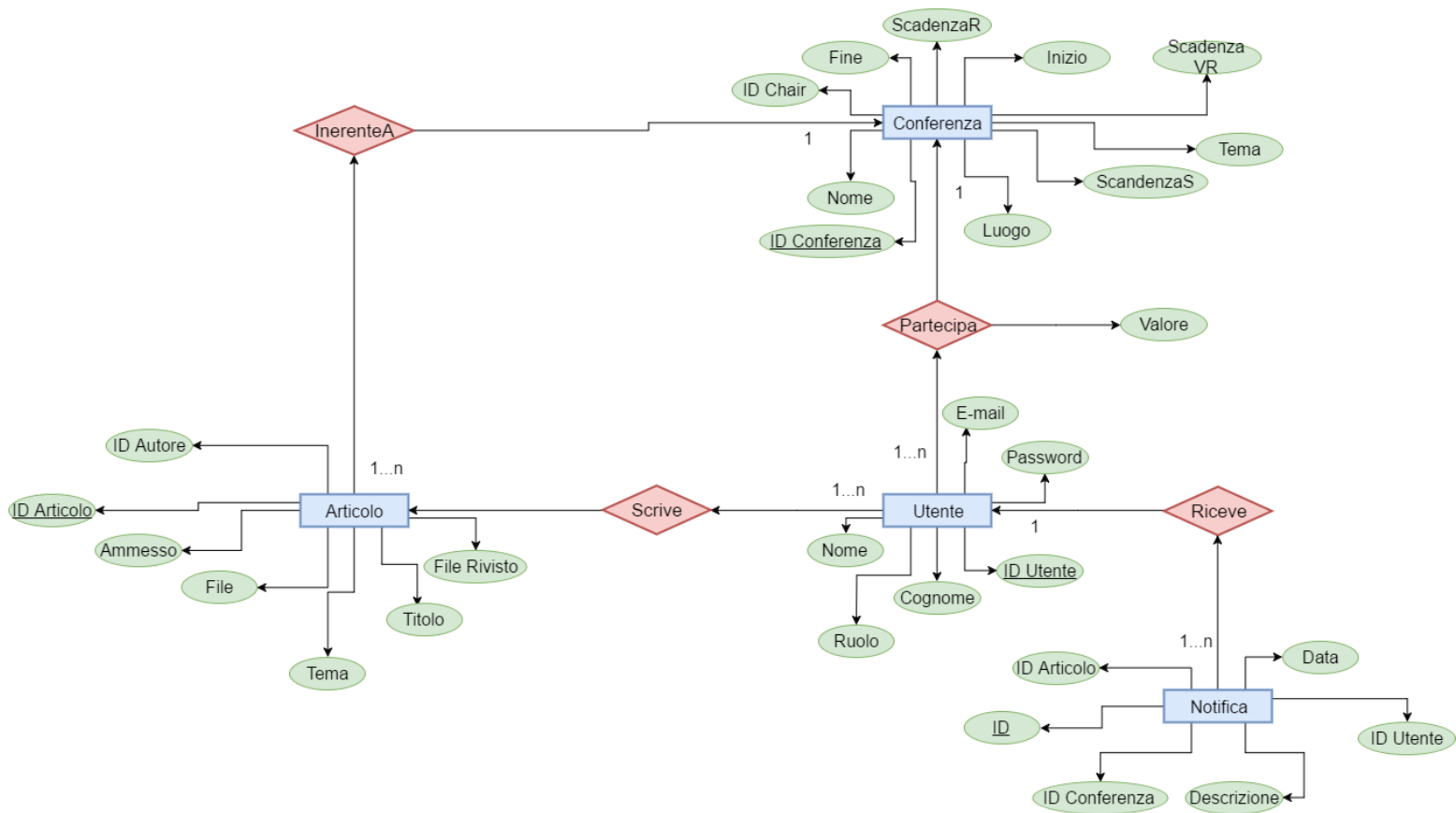
Il package è così organizzato:

- \* `ArticoloClass`
- \* `ConferenzaClass`
- \* `DataBaseManagement`
- \* `EmailClass`
- \* `NotificaClass`
- \* `RecensioneClass`
- \* `UtenteClass`

Particolare importanza ricopre la classe `DataBaseManagement` che implementando il driver `JDBC` rappresenta l'interfaccia attraverso la quale il sistema comunica col database `MySQL`. `EmailClass` implementa le `control` necessarie all'invio di email come notifica a particolari eventi.

## Database

## Diagramma ER



## Descrizione Tabelle

Di seguito una breve descrizione testuale della composizione delle varie tabelle seguita da una descrizione approfondita per ogni tabella.

### Descrizione Testuale

Tramite una descrizione verbale possiamo individuare le varie relazioni che sussistono tra le entità del sistema (colorate così: **Entità**, **Attributo**, **relazione**).

Una **conferenza** ha un **nome**, un **tema**, un **luogo**, una **data d'inizio** e una **data di fine**, un **numero massimo di partecipanti**, una **scadenza per la sottomissione degli articoli**, una **scadenza per effettuare la recensione degli articoli** e una **scadenza per la sottomissione degli articoli corretti**.

Un **utente** ha un **id**, un **nome**, un **cognome**, una **email** ed una **password**, in base all'id è possibile risalire al ruolo che occupa all'interno della conferenza.

Il Chair **crea** la conferenza e il comitato di programma (PC) **invitando** i revisori.

Il Chair potrà attribuire uno o più articoli da revisionare a ogni revisore che può decidere se **rifiutare** o **proporre** un sotto-revisore al posto suo che dovrà essere **confermato** dal Chair.

Nella fase di sottomissione degli articoli ogni autore **iscritto** potrà caricare il suo articolo con un **titolo**, un **tema** ed il **file allegato** che verrà contrassegnato con un **id**. Alla scadenza della data di sottomissione degli articoli il Chair assegnerà gli articoli ed il Comitato giudicherà gli articoli degli utenti tramite un **commento**, un **commento riservato altri revisori** e una **valutazione ponderata**.

Alla scadenza della fase di revisione, gli articoli ammessi e **partecipanti** alla conferenza, verranno contrassegnati come **ammesso** e l'autore avrà la possibilità di sottomettere una **versione corretta** dell'articolo.

## Tabella CONFERENZE

Nome Campo	Tipo	Vincoli	Descrizione
idConferenza	INT	PK, NOT NULL, AUTO_INCREMENT	id univoco assegnato ad ogni nuova conferenza
idChair	INT	NOT NULL	idUserente del creatore della conferenza
nome	VARCHAR(255) NOT NULL		Stringa che contiene il nome assegnato alla conferenza
tema	VARCHAR(255) NOT NULL		Stringa che contiene il tema della conferenza
luogo	VARCHAR(255) NOT NULL		Stringa che contiene il luogo in cui si terrà della conferenza
			Numero massimo di



numeroArticoli	INT	NOT NULL	articoli ammessi alla conferenza
inizio	DATE		Data di inizio della conferenza
fine	DATE		Data di fine conferenza
scadenzaSottomissione	DATE		Termine della fase di sottomissione degli articoli
scadenzaReview	DATE		Termine della fase di recensione degli articoli
scadenzaSottomissioneRivisti	DATE		Termine della fase di sottomissione degli articoli corretti

## Tabella UTENTI

Nome Campo	Tipo	Vincoli	Descrizione
idUtente	INT	PK, NOT NULL, AUTO_INCREMENT	id univoco assegnato ad ogni utente
email	VARCHAR(255)	NOT NULL	Stringa contenente l'email dell'utente
nome	VARCHAR(255)	NOT NULL	Stringa contenete il nome dell'utente
cognome	VARCHAR(255)	NOT NULL	Stringa contente il cognome dell'utente
password	VARCHAR(255)	NOT NULL	Stringa contente la password dell'utente

## Tabella ARTICOLI

Nome Campo	Tipo	Vincoli	Descrizione
		PK, NOT NULL,	

idArticolo	INT	AUTO_INCREMENT	id univoco assegnato ad ogni articolo
idUtente	INT	NOT NULL	idUtente dell'utente che sottomette l'articolo (autore)
titolo	VARCHAR(255)	NOT NULL	Stringa contenente il titolo dell'articolo
tema	VARCHAR(255)	NOT NULL	Stringa contenente il tema dell'articolo
file	VARCHAR(255)	NOT NULL	Stringa contenente il PATH del file
file_rivisto	VARCHAR(255)		Stringa contenente il PATH del file rivisto, qualora l'articolo fosse ammesso alla conferenza
ammesso	BOOLEAN	DEFAULT=false	Booleano che indica se un articolo è stato ammesso alla conferenza

## Tabella RECENSIONI

Nome Campo	Tipo	Vincoli	Descrizione
idRecensione	INT	PK, NOT NULL, AUTO_INCREMENT	id univoco assegnato ad ogni recensione
idArticolo	INT	NOT NULL	idArticolo dell'articolo oggetto della recensione
idRecensore	INT	NOT NULL	idUtente del recensore
votazione	DOUBLE	NOT NULL	Voto assegnato all'articolo, media ponderata di tre criteri per il livello di competenza del recensore nell'ambito
commento	VARCHAR(255)		Stringa contente il commento pubblico, viene inviato anche all'autore tramite email
commentoRiservato	VARCHAR(255)		Stringa contente il commento riservato al Chair

## Tabella NOTIFICHE

Nome Campo	Tipo	Vincoli	Descrizione
		PK, NOT NULL,	

id	INT	AUTO_INCREMENT	id univoco
idConferenza	INT		idConferenza della conferenza per cui la notifica è generata
idUtente	INT		idUtente dell'utente che genera la notifica
idArticolo	INT		idArticolo per cui si vuole notificare un evento (sottomissione, recensione) - opzionale -
descrizione	VARCHAR(255)		Stringa contenente una descrizione dell'evento che ha generato la notifica
data	DATE		Data in cui la notifica è stata generata

## Tabella AUTORI

Nome Campo	Tipo	Vincoli	Descrizione
id	INT	PK, NOT NULL, AUTO_INCREMENT	id univoco
idUtente	INT		idUtente dell'utente che partecipa come autore
idArticolo	INT		idArticolo dell'articolo sottomesso dall'autore

## Tabella COMITATO

Nome Campo	Tipo	Vincoli	Descrizione
id	INT	PK, NOT NULL, AUTO_INCREMENT	id univoco
idUtente	INT	NOT NULL	idUtente del membro del comitato
idArticolo	INT		idArticolo dell'articolo assegnato
idRecensione	INT		idRecensione della recensione sottomessa per l'articolo assegnato

# Accesso al Sistema

---

L'accesso al sistema è permesso tramite la finestra di login, dove l'utente deve inserire la propria email e la propria password, come specificato nel caso d'uso n°1 del RAD. CMS si preoccuperà di effettuare i relativi controlli. La logica di accesso al sistema si occupa anche di effettuare i controlli sul ruolo che l'utente ricopre nella conferenza: un utente autenticato è quindi totalmente identificato dopo l'accesso al sistema.