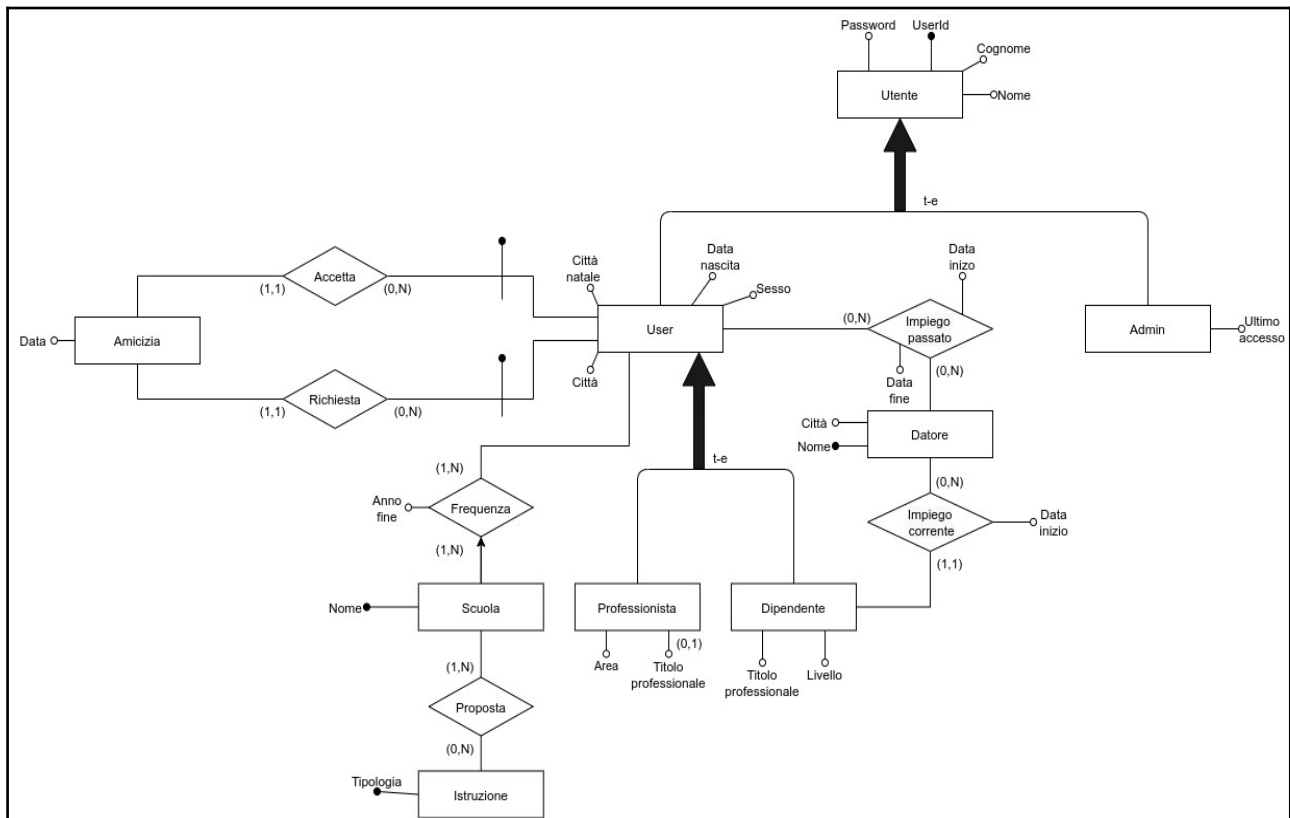


## Progettazione Concettuale

La piattaforma X memorizza gli utenti. Esistono due tipi di utenti. Ogni utente deve essere un user o un admin. Tutti gli utenti hanno un userId, una password, nome e cognome. Un admin ha anche la data e l'ora dell'ultimo accesso alla piattaforma. Un user ha anche un sesso, una data di nascita, una città attuale e una città natale e più interessi. La piattaforma memorizza i datori di lavoro. Ogni datore ha un nome univoco. Un user può essere correlato a molti datori, che a loro volta possono essere correlati a molti utenti. Nella relazione tra un user e un datore, è possibile memorizzare più qualifiche professionali. Un user può anche essere un libero professionista. La piattaforma archivia i nomi delle scuole e la loro tipologia. Ogni scuola ha un nome univoco. Tutte le scuole devono avere un tipologia (Istruzione superiore, Istruzione secondaria di secondo grado, Istruzione secondaria di primo grado, Istruzione primaria), che a sua volta può essere la tipologia di istruzione per più scuole. Un scuola può avere al suo interno più tipologia di istruzione. Un user può essere correlato a molte scuole, che a loro volta possono essere correlate a molti utenti. Nelle relazioni tra user e scuola, è possibile archiviare per ogni anno la data di fine studi. Ogni user può inviare più richieste di amicizia. Un'amicizia è determinata unicamente dal userId dell'user che l'ha richiesta e del userId dell'user che l'ha accettata. L'amicizia non è sempre reciproca... solo perché A è amica di B, questo non implica che B sia amica di A.

## Diagramma E-R



### Dizionario dei dati (entità)

Entità	Descrizione	Attributi	Identificatore
Utente	Insieme degli User e degli Admin	UserId, Password, Cognome, Nome	UserId
Admin	Gestori della piattaforma	UserId, Password, Cognome, Nome, Ultimo accesso	UserId
User	Iscritti della piattaforma, utenti regolari	UserId, Password, Cognome, Nome, Città natale, Data nascita, Sesso, Città	UserId
Professionista	Libero Professionista	Titolo professionale, Area	UserId
Dipendente	Lavoratore dipendente	Titolo Professionale, Livello	UserId
Datore	Ente presso cui i dipendenti lavorano o hanno lavorato	Nome, Città	Nome
Scuola	Scuola frequentata o che sta frequentando l'iscritto alla piattaforma	Nome	Nome
Istruzione	Tipologia di Istruzione fornita dalla scuola: primaria, secondaria ecc.	Tipologia	Tipologia
Amicizia	Elenco degli amici dell'iscritto alla piattaforma	Data	UserId

### Dizionario dei dati (relationship)

Relazioni	Descrizione	Componenti	Attributi
Impiego passato	Precedenti occupazioni lavorative	User, Datore	Data inizio, Data fine
Impiego corrente	Attuale occupazione lavorativa	Datore, Dipendente	Data inizio
Frequenza	Scuole frequentate dall'iscritto alla piattaforma	User, Scuola	Anno fine

Proposta	Tipologia di istruzione di proposta dalla scuola	Scuola, Istruzione	
Richiesta	Richieste di amicizia da accettare o rifiutare	User, Amicizia	
Accetta	Richieste di amicizia accettate	User, Amicizia	

## Vincoli non esprimibili

*Requisiti 1:* ...**Accedere** a una generica piattaforma X tramite la schermata di accesso. Tutti gli Utenti sono identificabili univocamente dalla loro **userId**. Fornire un **userId** e **password** valida, la combinazione di entrambi registrerà l'utente nel sistema. Fornire delle credenziali di accesso non valide dovrebbe visualizzare un messaggio di errore e riportare l'utente alla schermata di accesso.

*Requisiti 2:* ...Utenti che sono nuovi della piattaforma devono prima registrarsi. Un bottone per la **registrazione** è previsto direttamente nella pagina di login. Cliccando su tale pulsante viene visualizzata la pagina di registrazione utente. Dopo che l'utente ha cliccato su **registra**, il sistema dovrebbe verificare che tutti i campi siano compilati, che l' **userId** non sia già in uso e che **password** e **conferma password** siano uguali.

*Requisiti 3:* ...Tutti gli utenti della piattaforma (eccetto gli amministratori) hanno un Profilo Utente contenente le proprie informazioni. Dopo che un nuovo utente si è **registrato**, dovrebbe essere portato immediatamente alla schermata **Modifica Profilo**. Il profilo include le seguenti informazioni di base: **sex**, **compleanno**, **città**, **città natale**.

*Requisiti 4:* ...Il profilo contiene anche informazioni sull'**istruzione** dell'utente. L'insieme delle **scuole** disponibili e della loro **tipologia di istruzione** è gestito dalla piattaforma. Un utente può avere un numero qualsiasi di scuole associate al proprio profilo e può anche fornire una **data di fine carriera scolastica** per ogni scuola frequentata.

*Requisiti 5:* ...Il profilo contiene anche informazioni **professionali**. L'utente selezionerà il proprio **impiego** e quindi fornire la propria **qualifica professionale**.

*Requisiti 6:* ...Ci sono numerosi collegamenti nella schermata Visualizza Profilo:

- Cliccare su **Visualizza Amici** viene visualizzato l'elenco degli amici del suddetto utente.
- Cliccare su **Log Out** l'utente viene disconnesso dal sistema e viene visualizzata la schermata di accesso.

Inoltre l'utente attualmente connesso può:

- Cliccando su **Visualizza Richieste in Sospeso** e viene visualizzare l'elenco delle richieste di amicizia che non sono state ancora accettate o rifiutate.
- Cliccando su **Modifica Profilo** viene visualizzata la schermata di modifica del profilo che consente all'utente di apportare modifiche al proprio profilo.

*Requisiti 7:* ...Questa piattaforma così come ogni Social Network consente agli utenti di **cercare amici** e di mettersi in **contatto** con loro. Esistono diversi passaggi per stabilire una connessione con un nuovo amico:

- L'utente cerca un amico in base a diversi criteri inclusi nome, cognome, email.

- Quindi, l'utente invia una richiesta di amicizia a un altro utente con cui desidera mettersi in contatto.
- Infine, l'altro utente riceve la richiesta di amicizia e la può accettare o rifiutare.

*Requisiti 8:* ...Cliccando su **Visualizza Amici** nella schermata **Visualizza Profilo** viene visualizzato l'elenco degli amici per tale utente.

*Requisiti 9:* ...Tutti gli **utenti** sono identificati in modo univoco dal proprio **userId**. Fornendo un indirizzo **userId** e un **password** valida, l'utente accederà al sistema...

*Requisiti 10:* ...Tutti gli **utenti** (tranne gli **amministratori**) hanno un profilo contenente informazioni di base di loro...

*Requisiti 11:* ...Gli **admin** possiedono informazioni simili agli user come **userId**, **password**, **nome** e **cognome**... ma non hanno un profilo completo e non possono neanche compiere alcune azioni come inviare e ricevere richieste di amicizia. Un utente deve essere un **admin** o un **user**, ma non entrambi. La piattaforma tiene inoltre traccia dell'**ultima accesso** al sistema da parte di un **admin**.

*Requisiti 12:* ...Un elenco di **scuole**, dal quale l'utente può selezionare, viene gestito nel sistema. Supponiamo che tutti i nomi delle scuole siano univoci. Un utente può avere un numero qualsiasi di scuole associate e può fornire una **anno di fine** studi per ogni scuola.

*Requisiti 13:* ...Ogni scuola deve avere una **tipologia**. Esistono 4 tipi possibili: Istruzione superiore, Istruzione secondaria di secondo grado, Istruzione secondaria di primo grado, Istruzione primaria... Dovrebbe essere possibile per l'amministratore del database aggiungere nuove tipologie. A sua volta può essere la tipologia di istruzione per più scuole. Un scuola può avere al suo interno più tipologia di istruzione, ad esempio primaria e secondaria di primo grado.

*Requisiti 14:* ...Gli amministratori sono responsabili della gestione dell'elenco delle **datori**. Supponiamo che tutti i **datori** abbiano un **nome** univoco. Una user può essere stato impiegato in periodi diversi per la stessa azienda. Un user può anche essere un libero **professionista**.

*Requisiti 15:* ...La **qualifica professionale** non è gestito dall'amministratore e può essere un qualsiasi valore fornito dall'utente. Un user può contenere più lavori e lo stesso lavoro può perfino apparire più volte a condizione che la **qualifica professionale** sia diverso in ogni caso.

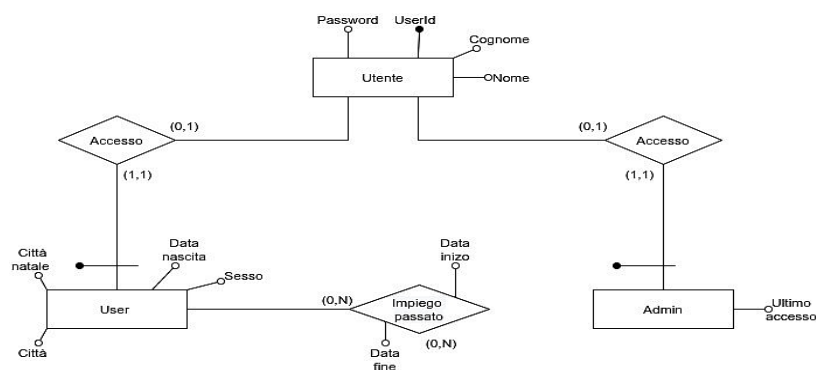
*Requisiti 16:* ...Un **amicizia** può non essere reciproca. Viene registrato la **data** in cui un richiesta di amicizia viene accettata, non quando tale richiesta viene inviata.

## Progettazione logica

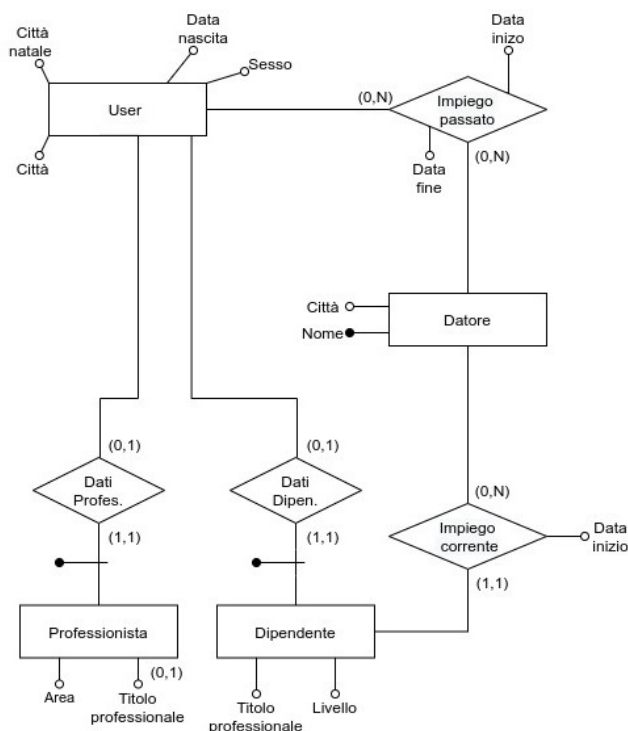
### Ristrutturazione dello schema E-R

#### Eliminazione delle gerarchie

Andiamo a sostituire la generalizzazione con associazioni

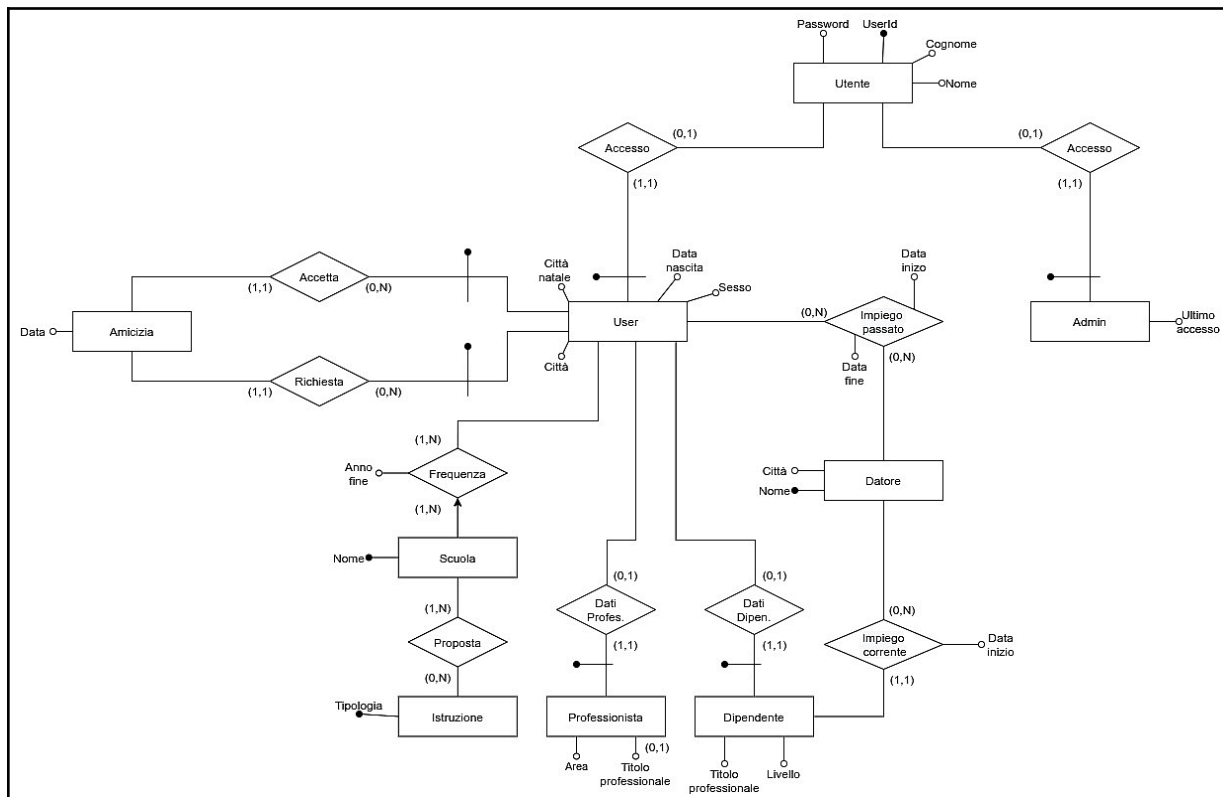


- Si introduce una relazione uno-a-uno fra l'entità padre e ciascuna entità figlia
- Si va ad inserire il vincolo che per ogni istanza dell'entità padre può partecipare solo ad una relazione di legame con le entità figlie
- Dato che la generalizzazione è totale ogni istanza dell'entità padre partecipa ad una sola delle relazioni di legame con le figlie



- I due sottotipi hanno attributi specifici
- Lasciamo i due sottotipi con entità e usiamo due relazioni per evitare di avere entità con troppi attributi e/o con attributi con valori nulli

## Schema E-R ristrutturato



## Traduzione verso il modello relazionale

Il Diagramma E-R illustrato è stato tradotto nel seguente schema relazionale; il/i campo/i che è/sono chiavi primarie vengono evidenziati da un sottolineatura. Un \* indica la possibilità di avere valori nulli per gli attributi.

Utente (UserId, Password, Nome, Cognome)

Admin (UserId, Ultimo\_accesso)

User (UserId, Data\_nascita, Sesso, Città, Città\_nascita)

Scuola (Nome)

Istruzione (Tipologia)

Proposta (Scuola, Istruzione)

Frequenza (UserId, Scuola, Anno\_fine\*)

Amicizia (UserId, Amico UserId, Data\*)

Datore (Nome, Città)

Impiego\_Passato (UserId, Datore, Data\_inizio, Data\_fine)

Professionista (UserId, Area, Titolo\_professionale\*)

Dipendente (UserId, Datore, Livello, Titolo\_professionale, Data\_inizio)

Nello schema logico sopra descritto fra le tabelle esistono i seguenti vincoli di integrità referenziale:

Admin.UserId → Utente.UserId  
User.UserId → Utente.UserId  
Proposta.Scuola → Scuola.Nome  
Proposta.Istruzione → Istruzione.Tipologia  
Frequenza.UserId → User.UserId  
Frequenza.Scuola → Scuola.Nome  
Amicizia.UserId → User.UserId  
Amicizia.Amico\_UserId → User.UserId  
Impiego\_Passato.UserId → User.UserId  
Impiego\_Passato.Datore → Datore.Nome  
Professionista.UserId → User.UserId  
Dipendente.UserId → User.UserId  
Dipendente.Datore → Datore.Nome