

## **PROGETTO:** Gestione di un social/community di videogiochi

### **1. Descrizione e Specifiche**

Si vuole progettare una Basi di Dati della Gestione di un social/community sui videogiochi

- In questo social sono coinvolti degli **UTENTI**
- Un utente è rappresentato da un **nome utente**, una **mail** e una **Password**, il **numero di software che possiede**, dal proprio **saldo** ed un **id** che lo identifica univocamente
- Gli utenti hanno una **LISTA AMICI** in cui vi sono altri utenti. Un utente può avere da 0 a n amici
- la lista amici è rappresentata dagli **id delle persone che sono amici tra di loro**
- Ogni utente può creare o partecipare a più **DISCUSSIONI**, ma ogni discussione è creata da uno e un solo utente
- La discussione è rappresentata da un **oggetto della chat**, una **data**, ed un **id** che la identifica univocamente
- In ogni discussione possono essere presenti da 0 a n **COMMENTI**
- Un commento è presente una e una sola volta in una discussione
- Un commento è rappresentato dal suo **contenuto**, dalla **data** e dall'**ora** di quando è stato postato e dall'id dell'utente che lo ha postato
- Un utente può scrivere molteplici commenti
- Un commento è scritto da uno e un solo utente
- Ogni utente possiede una **LIBRERIA** che contiene da 0 a n **SOFTWARE**
- La libreria è collegata all'id dell'utente che la possiede
- Ogni libreria può contenere da 0 a n software
- Ogni software può essere contenuto in 0 o n librerie
- I software si dividono in **VIDEOGIOCHI** e **UTILITY**

- Ogni software è rappresentato dal **nome**, dal **costo**, dallo **sviluppatore** e da un **id** che lo identifica univocamente
- I videogiochi sono rappresentati oltre che dagli attributi del software anche dall'attributo **genere**
- I Software di Utility sono rappresentati oltre che dagli attributi del software anche dall'attributo **descrizione**
- Ogni utente può aggiungere Software alla propria libreria, acquistandoli
- Ogni videogioco ha da n a m **TROFEI**
- Un trofeo è rappresentato da un **nome** e da un **id** che lo identifica univocamente
- Un utente può aver sbloccato molteplici trofei
- Un trofeo può essere stato sbloccato da molteplici utenti

## **2. Glossario dei termini**

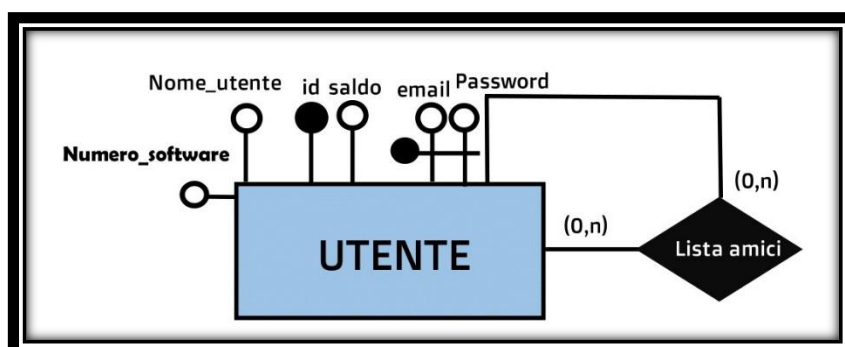
<b>Termine</b>	<b>Descrizione</b>	<b>Sinonimi</b>	<b>Termini Collegati</b>
Utente	Utente iscritto alla community		Chat, Libreria,Trofeo
Discussione	Discussione creata da un Utente		Utente
Commento	Commento scritto da un Utente		Utente, Discussione
Software	Software in vendita che può essere posseduto da un Utente e quindi presente nella sua libreria	Videogioco, Utility	Utente, Libreria

Libreria	Libreria di Software posseduti da un Utente		Utente, Software, Videogiochi, Utility
Videogioco	Videogioco in vendita che può far parte di una o più librerie di Utenti	Software	Libreria, Trofeo, Utente
Utility	Software di utility in vendita che può Far parte di una o più librerie di un Utente	Software	Libreria,Utenteo
Trofeo	Trofeo che fa parte di un Videogioco e che può essere sbloccato da un Utente		Videogioco, Utente

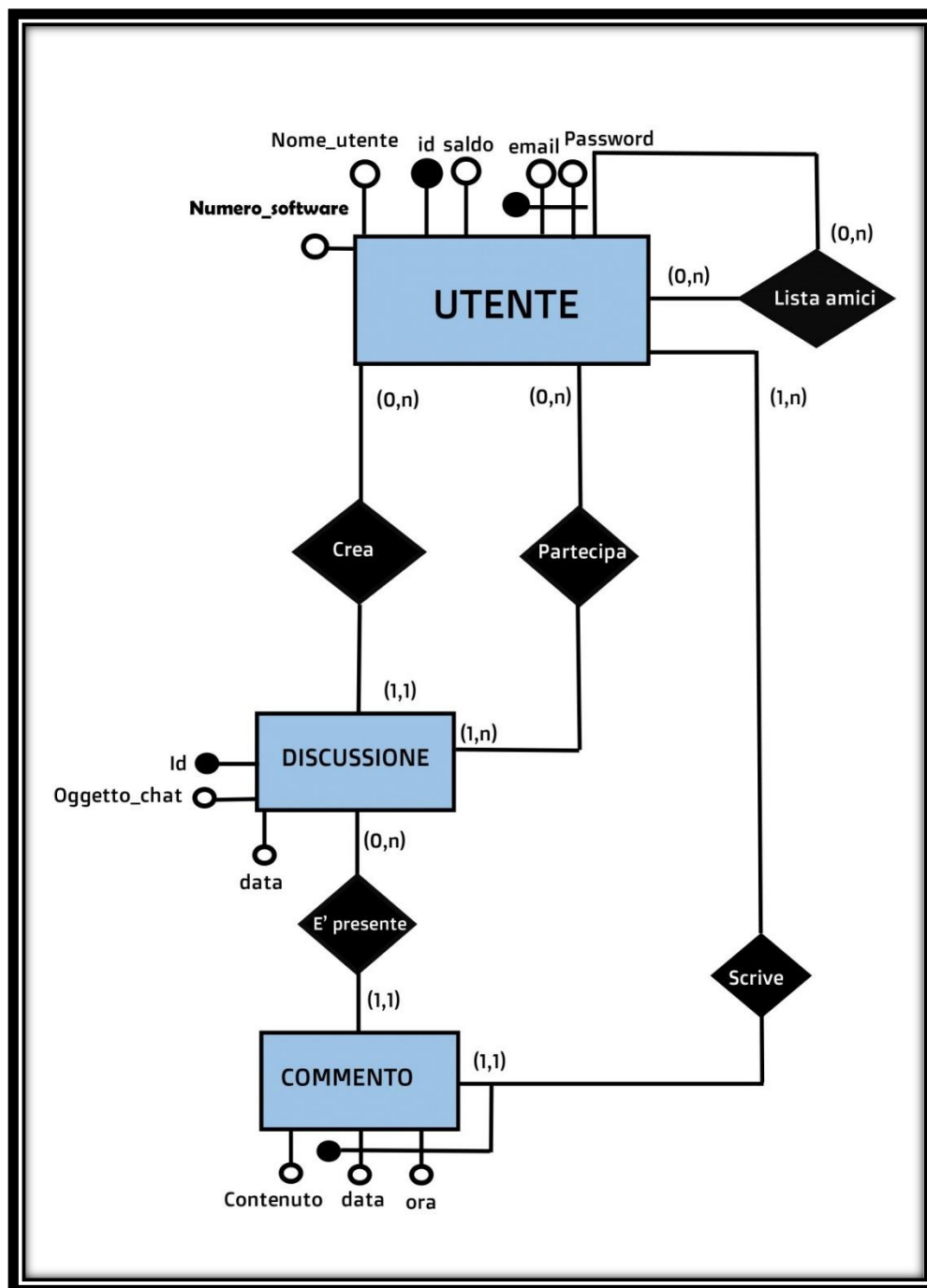
### 3. SCHEMA SCHELETRO

(Strategia Top-Down)

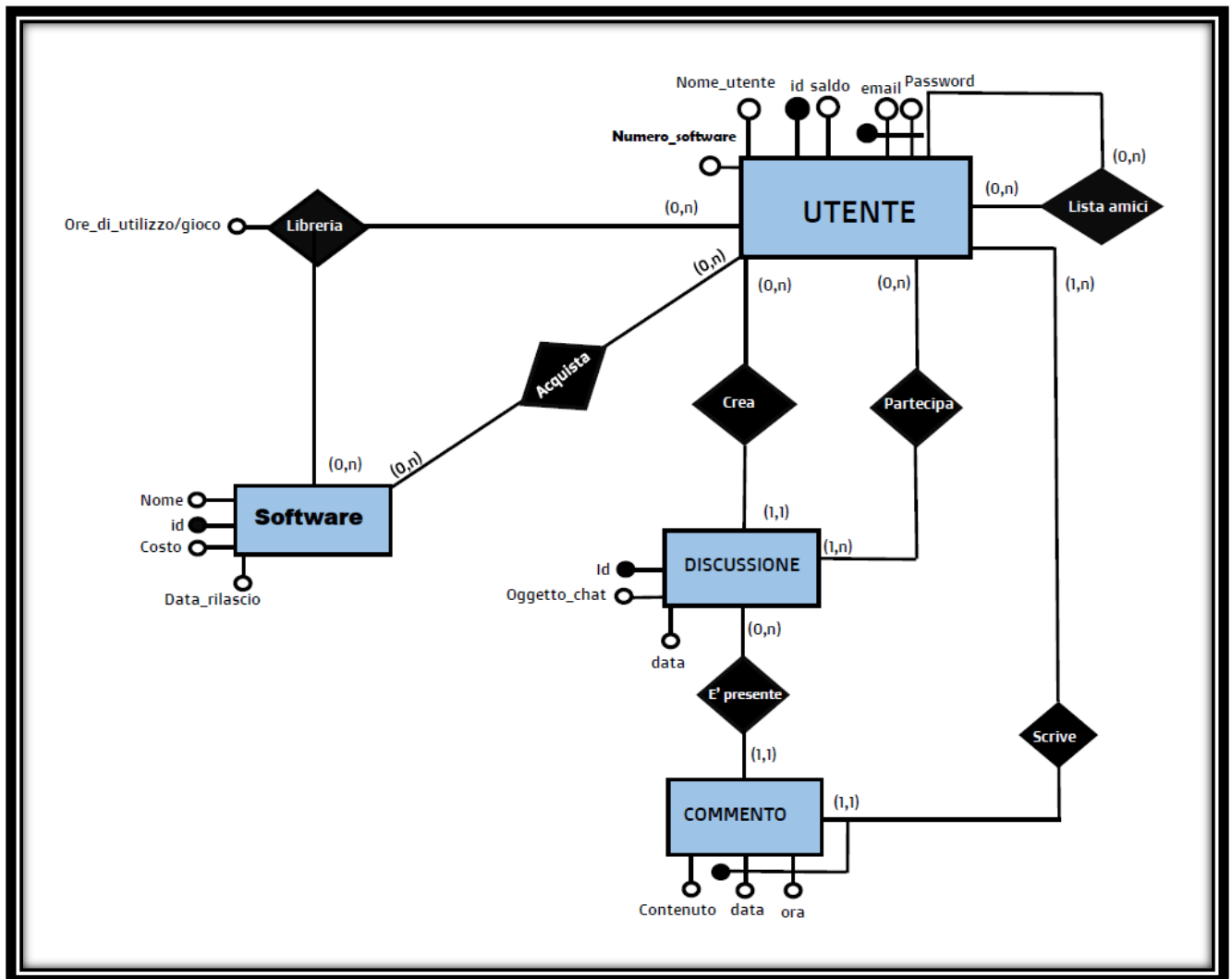
Implementazione Entità utente con associazione lista amici



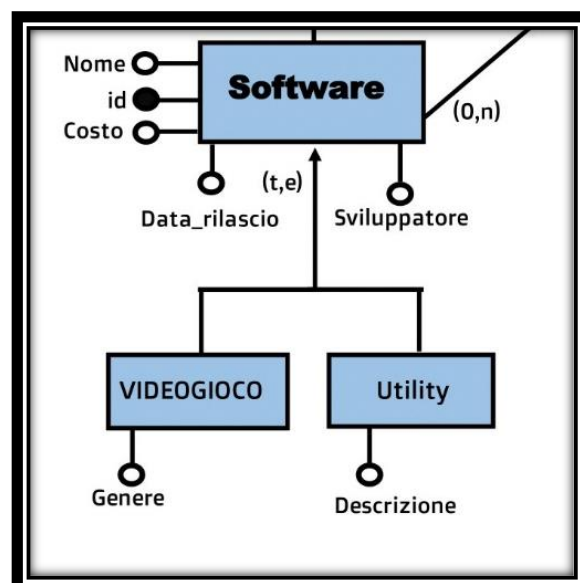
1. Raffinamento, implementazione discussioni e possibilità di aggiugnere commenti



2. Raffinamento, implementazione libreria di ogni utente a cui sono associati dei software

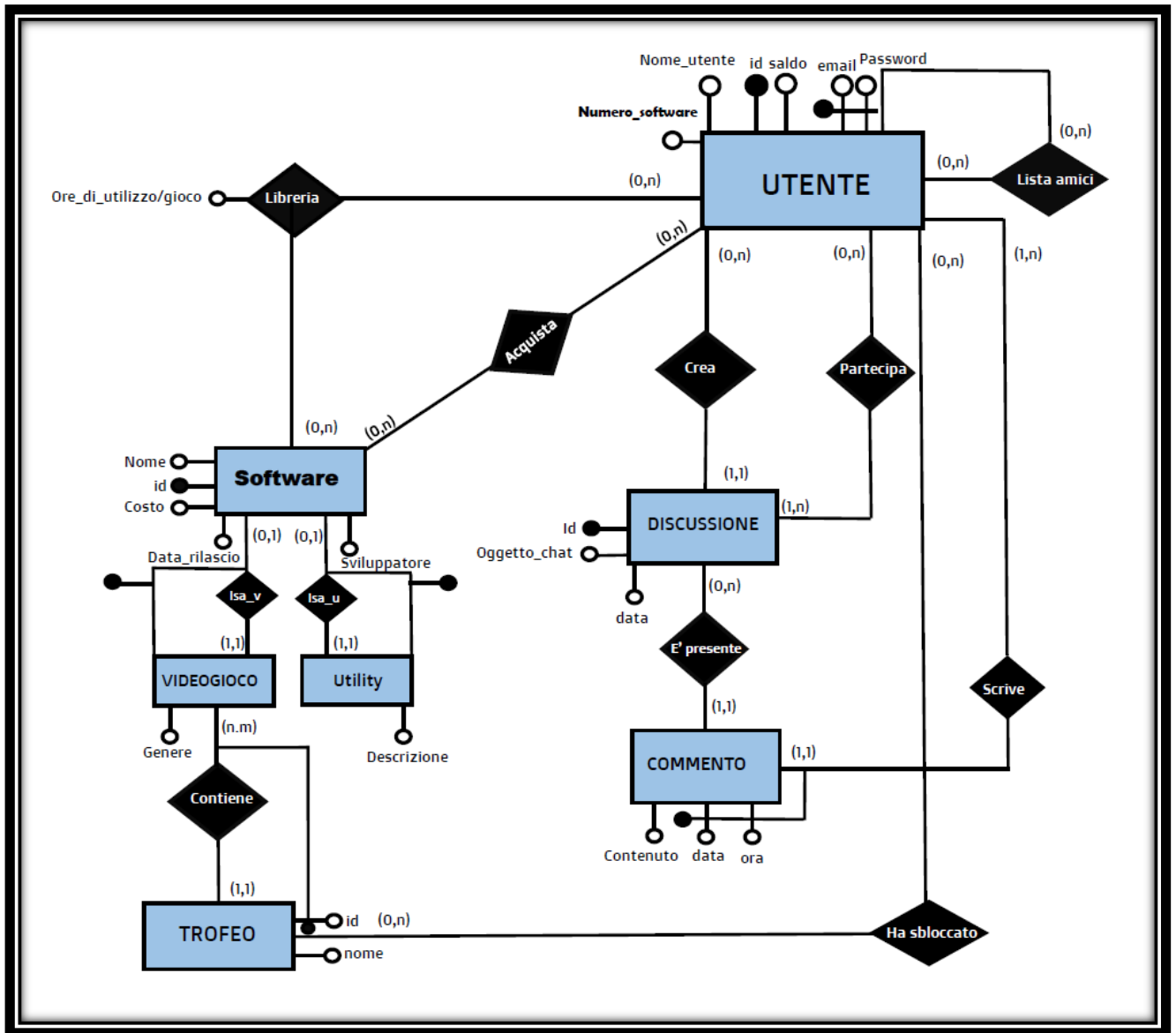


3. Raffinamento, introduzione della gerarchia sull'entità Software



## 4. Schema Concettuale

E' stato effettuato un passo di ristrutturazione, sono state eliminate le gerarchie con mantenimento delle Entità



## 5. Dizionario dei Dati – Entità

ENTITA'	ATTRIBUTI	IDENTIFICATORE
Utente	Nome_utente, Password, id	id
Discussione	Oggetto_chat, data, id	id
Commento	Contenuto, data, ora	Contenuto, data, ora, id_utente
Software	Nome, id, costo, data_rilascio, sviluppatore	id
Videogioco	Nome, id, costo, data_rilascio, sviluppatore, genere	id
Utility	Nome, id, costo, data_rilascio, sviluppatore, descrizione	id
Trofeo	id, nome	id,id_videogioco

## 6. Dizionario dei Dati – Relazioni

RELAZIONE	ENTITA' PARTECIPANTI	DESCRIZIONE
Lista amici	Utente	Un utente può avere dentro la propria lista amici altri utenti
Scrivo	Utente, Commento	Un utente scrive uno o più commenti
Crea	Utente, Discussione	Un utente crea una o più discussioni
Partecipa	Utente, Discussione	Un utente partecipa ad una o più discussioni
E' presente	Commento, Discussione	Un commento è presente in una discussione
Contiene	Videogioco, Trofeo	Un videogioco contiene dei trofei sbloccabili dagli utenti
Ha sbloccato	Utente, Trofeo	Un utente può sbloccare dei trofei
Acquista	Utente, Software	Un utente può acquistare dei software
Libreria	Utente, Software	Un utente possiede dei software



## 7. TABELLA DEI VOLUMI

CONCETTO	TIPO	VOLUME
Utente	E	500
Discussione	E	250
Commento	E	1500
Software	E	3000
Videogioco	E	2000
Utility	E	1000
Trofeo	E	20000

Lista amici	R	5000
Libreria	R	5000
Scrive	R	2500
Crea	R	250
Partecipa	R	500
E' presente	R	1500
Contiene	R	20000
Ha sbloccato	R	2500
Acquista	R	2500

## **8. Specifiche sulle Operazioni**

- 1) **Aggiungi amico:**  
Viene aggiunto un amico alla lista amici di un utente
  - 2) **Crea discussione:**  
Viene creata una nuova discussione da un utente
  - 3) **Inserisci commento:**  
Viene aggiunto un commento ad una discussione
  - 4) **Sblocca trofeo:**  
Un utente sblocca un trofeo
  - 5) **Acquista software:**  
Un utente acquista e quindi aggiunge un software alla sua libreria
  - 6) **Utente registrato:**  
Un nuovo utente si registra alla community
  - 7) **Aggiunta software:**  
Un nuovo software acquistabile viene aggiunto alla lista dei software
  - 8) **Calcolo numero software**  
Calcola il numero di software totali posseduti da un utente
  - 9) **Calcolo ore totali di ogni giocatore su tutti i software:**  
Calcolo il numero di ore totali che un utente ha accumulato
- Tavola delle frequenze:

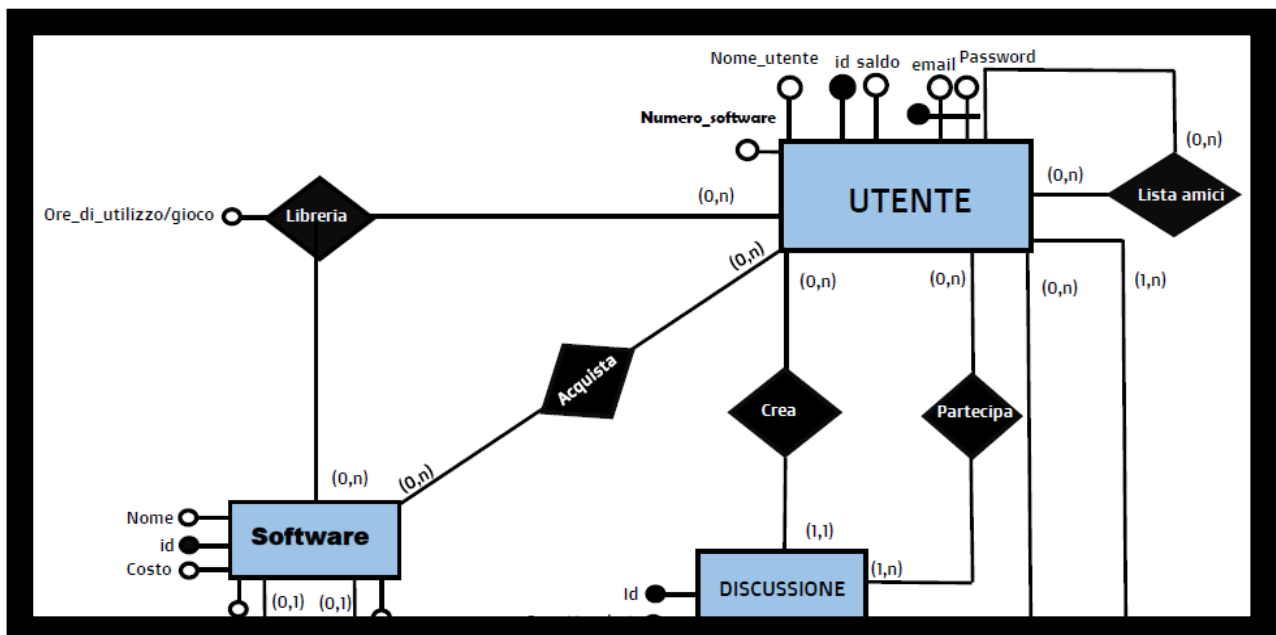
OP.	FREQUENZA	TIPO
1	50/giorno	I
2	300/giorno	I
3	100/giorno	I
4	20/giorno	I
5	30/giorno	I
6	10/giorno	I
7	10/mese	I
8	10/mese	B

9	1/mese	B
---	--------	---

- Op 1) 2L su Utente + 1S su Lista Amici =  $4L * 50 = 200L/giorno$   
 Op 2) 1L su Utente + 1S su discussione =  $3L * 300 = 900L/giorno$   
 Op 3) 1L su Utente + 1L su Discussione + 1S su Commento =  $4L * 100 = 400L/giorno$   
 Op 4) 1L su Utente + 1L su Libreria + 1L su Trofeo + 1S su Trofei Sbloccati =  $5L * 20 = 100L/giorno$   
 Op 5) 1L su Utente + 1L su Software + 1S su Utente (update del saldo) + 1S su libreria =  $5L * 30 = 150/giorno$
- Op 6) 1S su Utente =  $2L * 10 = 20L/giorno$   
 Op 7) 1S su Software =  $2L * 10 = 20L/mese$
- Op 9)  $x$  = numero di software posseduti da un utente  
 1L su Utente +  $xL$  su Libreria =  $x + 1 L /mese$

## 9. CALCOLO RIDONDANZE

Op 8) Studiamo se sia meglio mantenere l'attributo ridondante "Numero software" nell'entità Utente supponendo di voler calcolare il numero totale di software posseduti da un Utente



Con ridondanza: 1L su UTENTE = 500L/mese

Senza ridondanza: 1L su UTENTE + 10L su LIBRERIA =  $11L * 500 UTENTI = 5500L/mese$

In generale conviene mantenere la ridondanza.

## 11. PROCEDURE

1. Procedura per permettere ad un utente di acquistare un software, essa controllerà prima che il saldo dell'utente sia sufficiente per acquistare il gioco

```
1 BEGIN
2 declare s float;
3 declare c float;
4
5 SELECT utente.saldo INTO s
6 FROM utente
7 WHERE utente.id_utente=id_utenteIN;
8
9 SELECT software.costo INTO c
10 FROM software
11 WHERE software.id_software=id_softwareIN;
12
13 IF(c<=s) THEN
14 UPDATE utente
15 SET utente.saldo=s-c
16 WHERE utente.id_utente=id_utenteIN;
17
18 INSERT INTO libreria (id_utente,id_software,tempo_utilizzo) VALUES (id_utenteIN,id_softwareIN,0);
19 UPDATE utente
20 SET nsoftware=nsoftware+1
21 WHERE utente.id_utente=id_utenteIN;
22 END IF;
23 END
```

2. Procedura che aggiunge alla lista amici due utenti, inserisce pure la tupla "al contrario" in modo tale che se l'utente X è amico di Y risulti che Y è amico di X, prima ancora controlla che gli utenti siano differenti

```
1 IF (id1<>id2) THEN
2 INSERT INTO lista_amici VALUES (id1,id2),(id2,id1);
3 END IF
```

3. Procedura che aggiunge un trofeo alla lista dei trofei sbloccati di un utente ma solo se l'utente possiede il Videogioco a cui è collegato il trofeo

```
1 BEGIN
2 IF EXISTS (
3 SELECT *
4 FROM libreria
5 WHERE libreria.id_utente=id_utenteIN AND libreria.id_software = (
6 SELECT trofeo.id_software
7 FROM trofeo
8 WHERE trofeo.id_trofeo=id_trofeoIN
9 )
10
11 )
12 THEN
13 INSERT INTO trofei_sbloccati (id_trofeo, id_utente) VALUES (id_trofeoIN, id_utenteIN);
14 END IF;
15 END
```

4. Procedura che incrementa il tempo di utilizzo di un software posseduto da un Utente

```
1 UPDATE libreria
2 SET tempo_utilizzo = ADDTIME(tempo_utilizzo, '00:30:00')
3 WHERE id_utente = idutenteIN AND id_software = idsoftwareIN
```

5. Procedura per l'inserimento di un nuovo software acquistabile, questa procedura controlla che sia stato dato in input o il GENERE del videogioco o la DESCRIZIONE del software di Utility in modo tale che Genere e Descrizione non possano essere riempiti contemporaneamente e dunque che un software debba essere esclusivamente un Videogioco o un Software

```
1 BEGIN
2 IF (TRIM(GenereIN) = '' AND TRIM(DescrizioneIN) != '') THEN
3
4 INSERT INTO software (nome, costo, sviluppatore, descrizione)
5 VALUES (NomeIN, CostoIN, SviluppatoreIN, DescrizioneIN);
6
7 ELSE IF (TRIM(GenereIN) != '' AND TRIM(DescrizioneIN) = '') THEN
8
9 INSERT INTO software (nome, costo, sviluppatore, genere)
10 VALUES (NomeIN, CostoIN, SviluppatoreIN, GenereIN);
11
12 END IF;
13 END IF;
14 END
```

6. Procedura che calcola il tempo totale di un utente sommando il tempo trascorso in ogni software

```
1 SELECT libreria.id_utente, SEC_TO_TIME(SUM(TIME_TO_SEC(libreria.tempo_utilizzo)))
2 AS 'tempo totale'
3 FROM libreria
4 WHERE id_utente=id_utenteIN|
```

Di seguito la lista di tutte le procedure implementate:

<b>Acquisto_software</b>	PROCEDURE
<b>Aggiunta_Software</b>	PROCEDURE
<b>Aggiunta_trofeo</b>	PROCEDURE
<b>Calcolo_Ore_Totali</b>	PROCEDURE
<b>Creazione_discussione</b>	PROCEDURE
<b>Incrementa_Tempo_Utilizzo_Software</b>	PROCEDURE
<b>Inserimento_commento</b>	PROCEDURE
<b>Registrazione_nuovo_utente</b>	PROCEDURE
<b>aggiungiamico</b>	PROCEDURE
<b>sblocca_trofeo</b>	PROCEDURE
<b>Calcolo_Numero_Software</b>	PROCEDURE

## 10. Vincoli espressi con Trigger

1. Un commento non può essere scritto in data e ora antecedenti a data e ora attuali

```
1 CREATE TRIGGER `VerificaDataCommento` BEFORE INSERT ON
  `commento`
2 FOR EACH ROW IF (NEW.ora < CURRENT_TIMESTAMP || NEW.ora
  > CURRENT_TIMESTAMP ) THEN
3     SET NEW.ora = CURRENT_TIMESTAMP;
4     END IF
5
```

2. Un utente non può creare una discussione con data e ora antecedenti a data e ora attuali

```
1 CREATE TRIGGER `VerificaDataCreazioneDiscussione` BEFORE
  INSERT ON `discussione`
2 FOR EACH ROW IF (NEW.Data_creazione < CURRENT_DATE ||
  NEW.Data_creazione > CURRENT_DATE ) THEN
3     SET NEW.Data_creazione = CURRENT_DATE;
4     END IF
```

3. Il costo di un software non può essere negativo

```
1 CREATE TRIGGER `VerificaCosto` BEFORE INSERT ON
  `software`
2 FOR EACH ROW IF NEW.costo < 0 THEN
3     SET NEW.costo=0;
4     END IF
5
```



## 12. VISTE IMPLEMENTATE

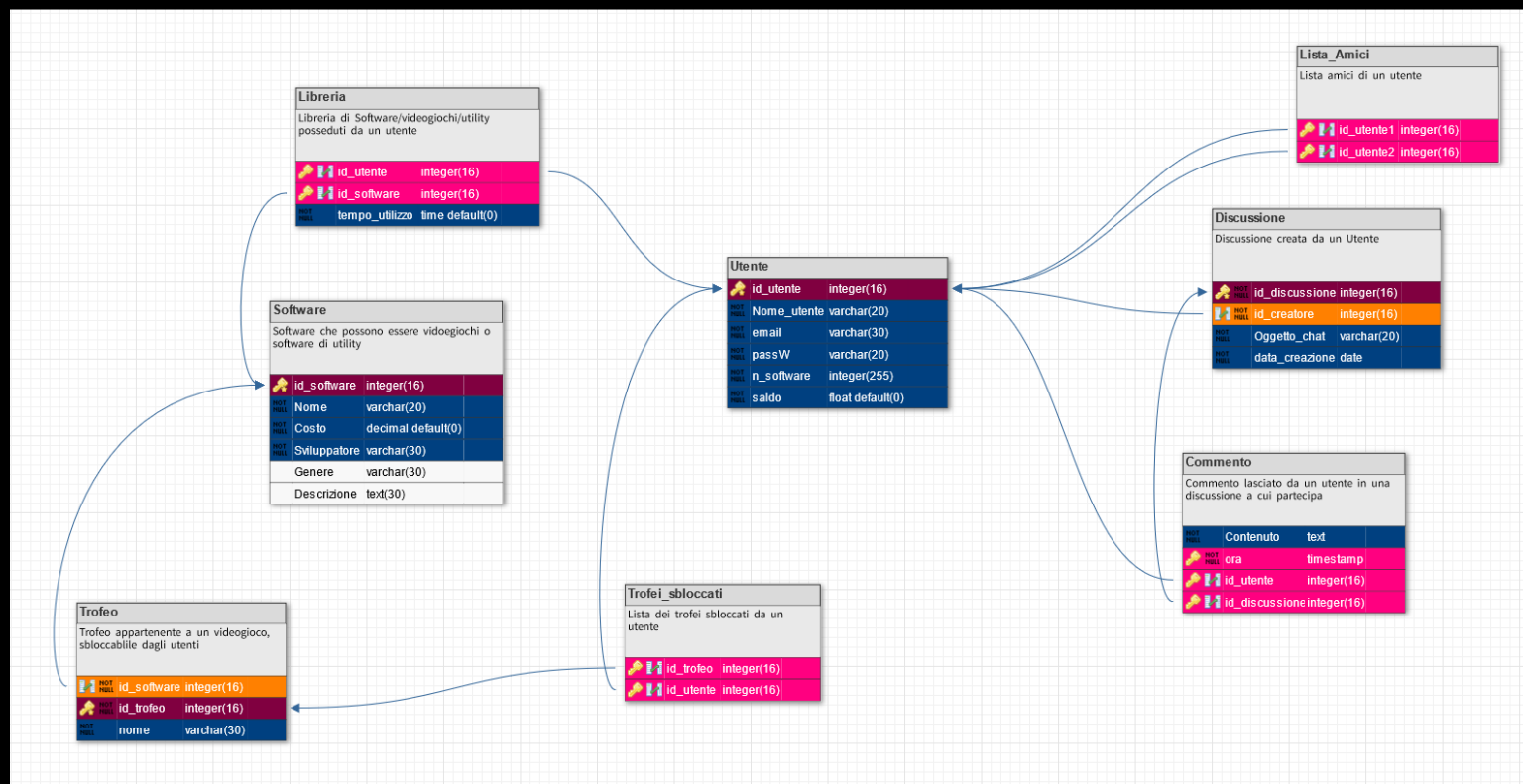
### 1. Vista di tutti i software che sono Videogiochi

```
1 CREATE VIEW Videogiochi AS
2 SELECT software.id_software, software.Nome AS 'Nome_Videogioco', software.Costo, software.Sviluppatore, software.Genere
3 FROM software
4 WHERE software.Genere IS NOT NULL and software.Descrizione IS NULL;
```

### 2. Vista di tutti i software che sono di Utility

```
1 CREATE VIEW utility_software AS
2 SELECT software.id_software, software.Nome AS 'Nome_Videogioco', software.Costo, software.Sviluppatore, software.Genere
3 FROM software
4 WHERE software.Genere IS NULL and software.Descrizione IS NOT NULL;
```

## 13. SCHEMA UML



## 14. Progettazione Logica

**Utente** (id\_Utente, Nome\_utente, email, Password, numero\_software, saldo);

**Libreria** (id\_software, id\_Utente, tempo\_utilizzo);

**Software** (id\_software, Nome, Costo, Sviluppatore, Genere, Descrizione);

**Trofeo** (nome, id\_trofeo, id\_software);

**Discussione** (id\_discussione, id\_creatore, Oggetto\_chat, data\_creazione);

**Commento** (Contenuto, timestamp, id\_utente, id\_discussione);

**Lista\_Amici** (id\_utente1, id\_utente2);

**Trofei\_Sbloccati** (id\_trofeo, id\_utente);

IN ALLEGATO TUTTI I FILE RELATIVI ALLA PROGETTAZIONE FISICA DEL DATABASE.