



SAPIENZA
UNIVERSITÀ DI ROMA

Basi di Dati, Modulo 2

Sapienza Università di Roma

Facoltà di Ing. dell'Informazione, Informatica e Statistica

Laurea in Informatica

Prof. Toni Mancini

<http://tmancini.di.uniroma1.it>

Progetto 20080401 (P.20080401)

TuTubi

Versione 2019-02-23

Indice

Indice	1
1 Introduzione	3
2 Specifica dei Requisiti	5
A Analisi Concettuale	7
A.1 Raffinamento dei Requisiti	9
A.1.1 Testo	9
A.1.2 Soluzione	10
A.2 Diagramma ER	13
A.2.1 Testo	13
A.2.2 Soluzione	14
A.3 Estensione del Diagramma ER	19
A.3.1 Testo	19
A.3.2 Soluzione	20
A.4 Estensione del Diagramma ER	25
A.4.1 Testo	25
A.4.2 Soluzione	26
A.5 Identificazione delle Entità e Definizione Informale di Vincoli Esterni	31
A.5.1 Testo	31
A.5.2 Soluzione	32

A.6	Diagramma UML degli Use-Case	39
A.6.1	Testo	39
A.6.2	Soluzione	40

1

Introduzione

Si vuole progettare e realizzare *TuTubi*, un sistema informatico da usarsi su un portale Web per la pubblicazione e condivisione di file video da parte degli utenti registrati. Il sistema deve permettere la memorizzazione dei video e la loro visualizzazione da parte degli utenti. Questi devono poter anche esprimere valutazioni e commenti.



2

Specifica dei Requisiti

TuTubi deve consentire la registrazione degli utenti, di cui interessa il nome e la data di iscrizione. Gli utenti registrati possono pubblicare video, visualizzare quelli disponibili, oltre che esprimere su di essi valutazioni e commenti testuali.

In particolare, di un video pubblicato da un utente interessa conoscere il titolo, la durata e la descrizione (oltre che il nome del file dove questo viene memorizzato da parte del sistema). Inoltre, di ogni video interessa conoscere la categoria (unica) a cui appartiene, oltre che un insieme di parole chiave dette *tag* (almeno una) che ne descrivano il contenuto in modo più strutturato. Delle categorie e dei *tag* interessa il nome.

TuTubi deve anche permettere ad un utente di pubblicare un nuovo video segnalando che si tratta di una risposta ad un video già esistente (utile per temi come la politica dove vi possono essere posizioni diverse che stimolano un dibattito). *TuTubi* deve mantenere tale informazione in modo consistente, garantendo che nessun utente possa pubblicare un video in risposta ad un video pubblicato da sé stesso.

TuTubi deve poi fornire un servizio di cronologia, ricordando la sequenza di tutti i video visionati da ogni singolo utente, con relativa data e ora. Inoltre, per ogni video interessa conoscere il numero complessivo di volte che è stato visionato.

Gli utenti di *TuTubi* devono inoltre avere la facoltà di esprimere una valutazione (un valore da 0 –pessimo– a 5 –ottimo) per ogni video che visionano. Tali valutazioni serviranno poi al sistema per promuovere i video più belli (cfr. seguito) e scovare i più brutti. Tuttavia, per evitare degenerazioni nel meccanismo delle votazioni, l'utente che ha pubblicato un video non può votarlo, mentre ogni altro utente può votarlo al più una volta, indipendentemente dal numero di volte che l'ha visionato. In ogni caso però deve essere impossibile per un utente votare un video che non ha mai visionato.

Inoltre, per favorire uno spirito di comunità tra gli utenti del servizio, si prevede che questi possano esprimere commenti testuali ai video che visionano (dei quali interessa anche data e ora). A differenza delle valutazioni, un utente può esprimere più commenti

per lo stesso video. Anche qui, non deve essere permesso ad un utente di scrivere commenti per video che non ha mai visionato.

Ogni utente può creare delle *playlist* personali, ovvero delle collezioni ordinate di video che gradisce vedere, oppure vuole condividere con altri utenti. Le playlist infatti (di cui interessa il nome e la data di creazione) possono essere pubbliche o private: solo le playlist pubbliche possono essere visualizzate dagli altri utenti. A tal fine, il sistema deve permettere ad ogni utente di ottenere le playlist pubbliche di un altro utente a sua scelta.

TuTubi deve permettere ad un utente di iscriversi, pubblicare nuovi video, creare e modificare le sue playlist, ed esprimere valutazioni e commenti sui video che visiona.

Inoltre, *TuTubi* deve consentire la ricerca di video: in particolare, data una categoria, un insieme di tag, ed un intero v tra 0 e 5, si vogliono restituire tutti i video disponibili di quella categoria che posseggono almeno uno tra i tag indicati, e che abbiano una valutazione media di almeno v (se un video non ha ancora alcuna valutazione, deve essere restituito comunque). *TuTubi* deve poi permettere di cercare, data una categoria, i video di quella categoria che hanno il numero maggiore di video in risposta, al fine di isolare le discussioni più animate tra gli utenti.

La redazione di *TuTubi* ha infine la facoltà di censurare dei video, ad esempio perché di contenuto coperto da copyright, osceno, ecc. Un video censurato non può essere né visionato, né votato, né commentato, né aggiunto ad alcuna playlist, né restituito come risultato di una ricerca. Un video, una volta censurato, non può tornare più visibile. Il motivo di una censura deve essere mantenuto nel sistema per usi interni.

Parte A

Analisi Concettuale

A.1

Raffinamento dei Requisiti

A.1.1 Testo

Raffinare la specifica dei requisiti eliminando inconsistenze, omissioni o ridondanze e producendo un elenco numerato di requisiti il meno ambiguo possibile.

A.1.2 Soluzione

1. Di ogni utente iscritto interessa:
 - 1.1. il nome
 - 1.2. la data di iscrizione
2. Di ogni video pubblicato da un utente interessa:
 - 2.1. il titolo
 - 2.2. la descrizione
 - 2.3. la durata (in secondi)
 - 2.4. il nome del file dove è memorizzato
 - 2.5. l'utente che lo ha pubblicato ([Req. 1.](#))
 - 2.6. la categoria a cui appartiene ([Req. 3.](#))
 - 2.7. i tag associati, almeno uno ([Req. 4.](#))
 - 2.8. il numero complessivo di volte in cui è stato visionato (calcolabile, dato [Req. 5.](#))
 - 2.9. conoscere se è un video in risposta ad un altro video. In questo caso interessa:
 - 2.9.1. il video a cui viene risposto, che non deve essere pubblicato dallo stesso utente del video di risposta
 - 2.10. conoscere se è un video censurato. Dei video censurati interessa mantenere una descrizione della ragione della censura.
3. Di ogni categoria di video interessa il nome
4. Di ogni tag associato a video interessa il nome
5. Per ogni utente interessa conoscere l'elenco di tutti i video che ha visionato. Per ogni visualizzazione interessa la data e l'ora.
6. Il sistema deve poter rappresentare valutazioni di video da parte degli utenti
 - 6.1. la valutazione è espressa come un intero tra 0 (pessimo) e 5 (ottimo)
 - 6.2. ogni utente può esprimere al più una valutazione per ogni video
 - 6.3. ogni utente può esprimere una valutazione per un video solo dopo averlo visionato ([Req. 5.](#))
 - 6.4. un utente non può esprimere una valutazione per un video pubblicato da egli stesso
7. Il sistema deve poter rappresentare commenti da parte degli utenti su video
 - 7.1. Di ogni commento interessa:

- 7.1.1. l'utente che lo ha scritto
 - 7.1.2. il testo associato
 - 7.1.3. il video oggetto del commento
 - 7.1.4. la data e l'ora
- 7.2. Un utente può scrivere commenti su un video solo se l'ha visionato ([Req. 5.](#))
- 8. Di ogni playlist interessa:
 - 8.1. l'utente che l'ha definita
 - 8.2. il nome
 - 8.3. la data di creazione
 - 8.4. la sequenza di video che contiene
 - 8.5. conoscere se è pubblica o privata
- 9. Il sistema deve fornire le seguenti funzionalità:
 - 9.1. gli utenti devono potersi iscrivere nel sistema, fornendo i dati necessari
 - 9.2. gli utenti iscritti devono poter pubblicare video, fornendo i dati necessari
 - 9.3. gli utenti iscritti devono poter creare e modificare playlist
 - 9.4. gli utenti (anche non iscritti) devono poter visionare video, se non censurati
 - 9.5. gli utenti iscritti devono poter valutare video, se non censurati
 - 9.6. gli utenti iscritti devono poter esprimere commenti su video, se non censurati
 - 9.7. gli utenti (anche non iscritti) devono poter ricercare video non censurati. In particolare:
 - 9.7.1. dati una categoria, un insieme di tag, ed un intero v tra 0 e 5, si vogliono restituire tutti i video non censurati di quella categoria che posseggono almeno uno tra i tag indicati, e che abbiano una valutazione media di almeno v (se un video non ha ancora alcuna valutazione, deve essere restituito comunque)
 - 9.7.2. data una categoria, si vogliono restituire i video non censurati di quella categoria che hanno il numero maggiore di video in risposta
 - 9.8. la redazione deve poter censurare video ([Req. 2.10.](#)).



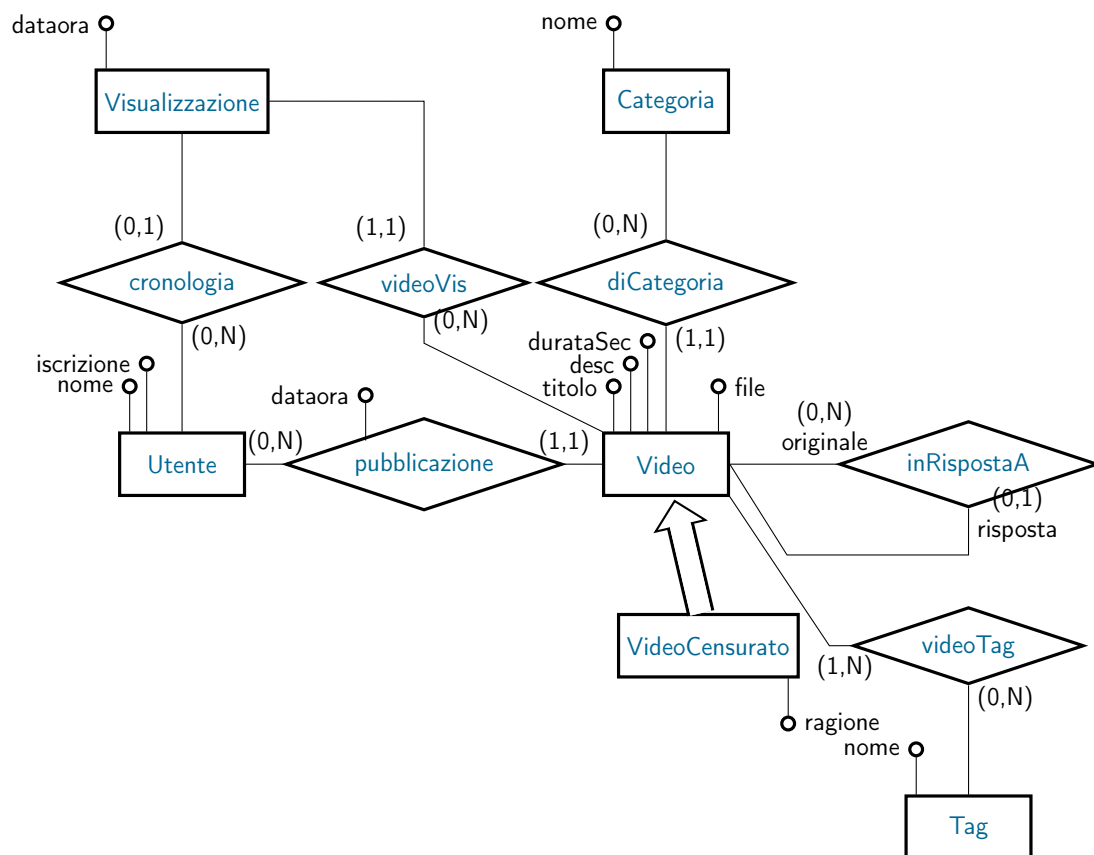
A.2

Diagramma ER

A.2.1 Testo

Proseguire la fase di Analisi Concettuale dei requisiti a partire dall'output della fase di raffinamento effettuata al passo A.1. In particolare, produrre il diagramma ER concettuale per l'applicazione ed il dizionario dei dati per modellare i seguenti requisiti: [Req. 1.](#), [Req. 2.](#), [Req. 3.](#), [Req. 4.](#), [Req. 5.](#)

A.2.2 Soluzione



Specifiche dei Dati

Entità **Utente**

Ogni istanza di questa entità rappresenta un utente registrato ([Req. 1.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome dell'utente (Req. 1.1.)
iscrizione	data		La data di iscrizione dell'utente al servizio (Req. 1.2.)

Entità **Video**

Ogni istanza di questa entità rappresenta un video caricato nel sistema ([Req. 2.](#))

attributo	dominio	molteplicità	descrizione
titolo	stringa		Il titolo del video (Req. 2.1.)
desc	stringa		La descrizione del video (Req. 2.2.)
durataSec	intero > 0		La durata del video in secondi (Req. 2.3.)
file	stringa		Il nome del file dove il video è memorizzato (Req. 2.4.)

Entità **VideoCensurato**

Ogni istanza di questa entità rappresenta un video censurato ([Req. 2.10.](#))

attributo	dominio	molteplicità	descrizione
ragione	stringa		La ragione della censura del video)

Entità **Visualizzazione**

Ogni istanza di questa entità rappresenta la visualizzazione di un video da parte di un utente ([Req. 5.](#))

attributo	dominio	molteplicità	descrizione
dataora	dataora		La data e l'ora della visualizzazione)

Entità *Categoria*

Ogni istanza di questa entità rappresenta una categoria ([Req. 3.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome della categoria

Entità *Tag*

Ogni istanza di questa entità rappresenta un tag ([Req. 4.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome del tag

Relationship *pubblicazione*

Ogni istanza di questa relationship lega un [Utente](#) ad uno dei [Video](#) che ha pubblicato ([Req. 2.5.](#))

attributo	dominio	molteplicità	descrizione
dataora	dataora		La data e l'ora della pubblicazione

Relationship *diCategoria*

Ogni istanza di questa relationship lega un [Video](#) alla sua [Categoria](#) ([Req. 2.6.](#))

Attributi: Nessuno

Relationship *videoTag*

Ogni istanza di questa relationship lega un [Video](#) ad uno dei suoi [Tag](#) ([Req. 2.7.](#))

Attributi: Nessuno

Relationship *inRispostaA*

Ogni istanza di questa relationship lega un [Video](#) (ruolo 'originale') al [Video](#) (ruolo 'risposta') a cui risponde ([Req. 2.9.](#))

Attributi: Nessuno

Relationship *cronologia*

Ogni istanza di questa relationship lega una [Visualizzazione](#) all'[Utente](#) che l'ha effettuata ([Req. 5.](#))

Attributi: Nessuno

Relationship [videoVis](#)

Ogni istanza di questa relationship lega una [Visualizzazione](#) al [Video](#) visualizzato ([Req. 5.](#))

Attributi: Nessuno



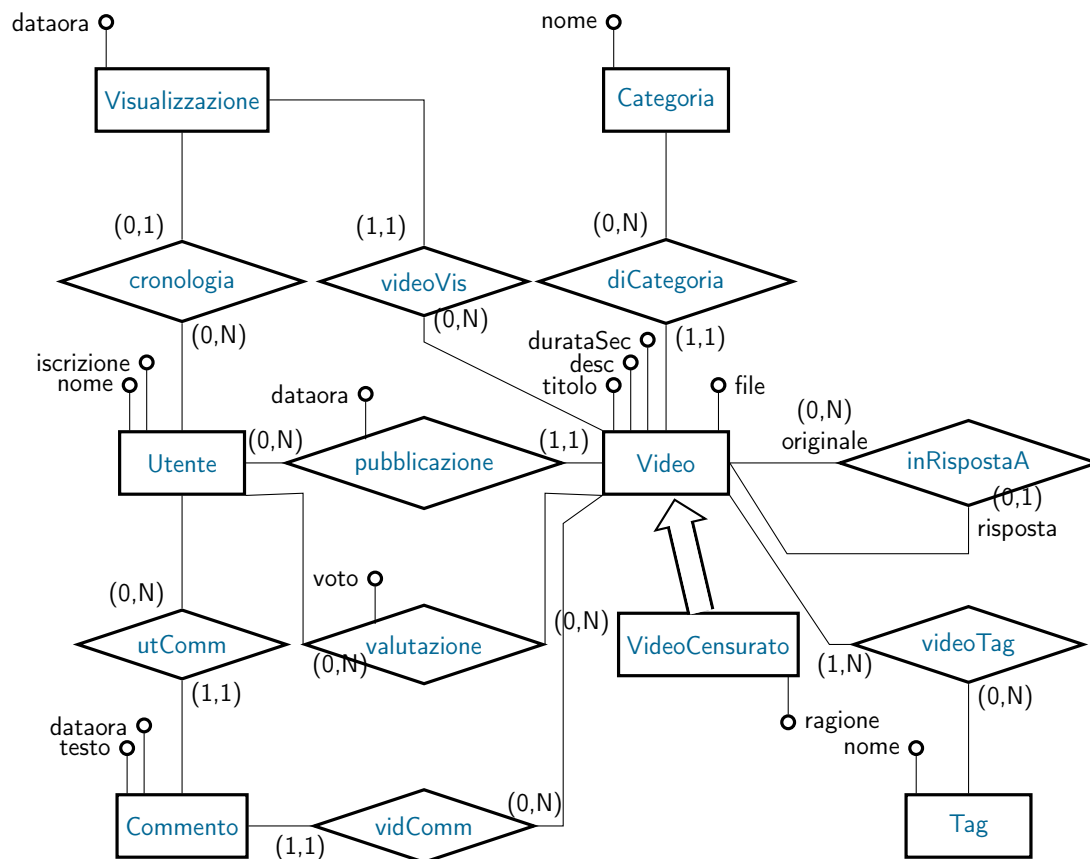
A.3

Estensione del Diagramma ER

A.3.1 Testo

Proseguire la fase di Analisi Concettuale dei requisiti estendendo il diagramma ER concettuale per l'applicazione ed il dizionario dei dati per modellare anche i seguenti requisiti: [Req. 6.](#), [Req. 7.](#)

A.3.2 Soluzione



Specifiche dei Dati

Entità **Utente**

Ogni istanza di questa entità rappresenta un utente registrato ([Req. 1.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome dell'utente (Req. 1.1.)
iscrizione	data		La data di iscrizione dell'utente al servizio (Req. 1.2.)

Entità **Video**

Ogni istanza di questa entità rappresenta un video caricato nel sistema ([Req. 2.](#))

attributo	dominio	molteplicità	descrizione
titolo	stringa		Il titolo del video (Req. 2.1.)
desc	stringa		La descrizione del video (Req. 2.2.)
durataSec	intero > 0		La durata del video in secondi (Req. 2.3.)
file	stringa		Il nome del file dove il video è memorizzato (Req. 2.4.)

Entità **VideoCensurato**

Ogni istanza di questa entità rappresenta un video censurato ([Req. 2.10.](#))

attributo	dominio	molteplicità	descrizione
ragione	stringa		La ragione della censura del video)

Entità **Visualizzazione**

Ogni istanza di questa entità rappresenta la visualizzazione di un video da parte di un utente ([Req. 5.](#))

attributo	dominio	molteplicità	descrizione
dataora	dataora		La data e l'ora della visualizzazione)

Entità [Categoria](#)

Ogni istanza di questa entità rappresenta una categoria ([Req. 3.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome della categoria

Entità [Tag](#)

Ogni istanza di questa entità rappresenta un tag ([Req. 4.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome del tag

Entità [Commento](#)

Ogni istanza di questa entità rappresenta un commento di un [Utente](#) ad un [Video](#) ([Req. 7.](#))

attributo	dominio	molteplicità	descrizione
testo	stringa		Il testo del commento
dataora	dataora		La data e l'ora del commento

Relationship [pubblicazione](#)

Ogni istanza di questa relationship lega un [Utente](#) ad uno dei [Video](#) che ha pubblicato ([Req. 2.5.](#))

attributo	dominio	molteplicità	descrizione
dataora	dataora		La data e l'ora della pubblicazione

Relationship [diCategoria](#)

Ogni istanza di questa relationship lega un [Video](#) alla sua [Categoria](#) ([Req. 2.6.](#))

Attributi: Nessuno

Relationship [videoTag](#)

Ogni istanza di questa relationship lega un [Video](#) ad uno dei suoi [Tag](#) ([Req. 2.7.](#))

Attributi: Nessuno

Relationship inRispostaA

Ogni istanza di questa relationship lega un **Video** (ruolo 'originale') al **Video** (ruolo 'risposta') a cui risponde (Req. 2.9.)

Attributi: Nessuno

Relationship cronologia

Ogni istanza di questa relationship lega una **Visualizzazione** all'**Utente** che l'ha effettuata (Req. 5.)

Attributi: Nessuno

Relationship videoVis

Ogni istanza di questa relationship lega una **Visualizzazione** al **Video** visualizzato (Req. 5.)

Attributi: Nessuno

Relationship valutazione

Ogni istanza di questa relationship rappresenta la valutazione di video espressa da un **Utente** (Req. 6.)

attributo	dominio	molteplicità	descrizione
voto	[0, 5]		La valutazione espressa, da 0 (pessimo) a 5 (ottimo)

Relationship utComm

Ogni istanza di questa relationship lega un **Commento** all'**Utente** che lo ha scritto (Req. 7.1.1.)

Attributi: Nessuno

Relationship vidComm

Ogni istanza di questa relationship lega un **Commento** al **Video** a cui si riferisce (Req. 7.1.3.)

Attributi: Nessuno



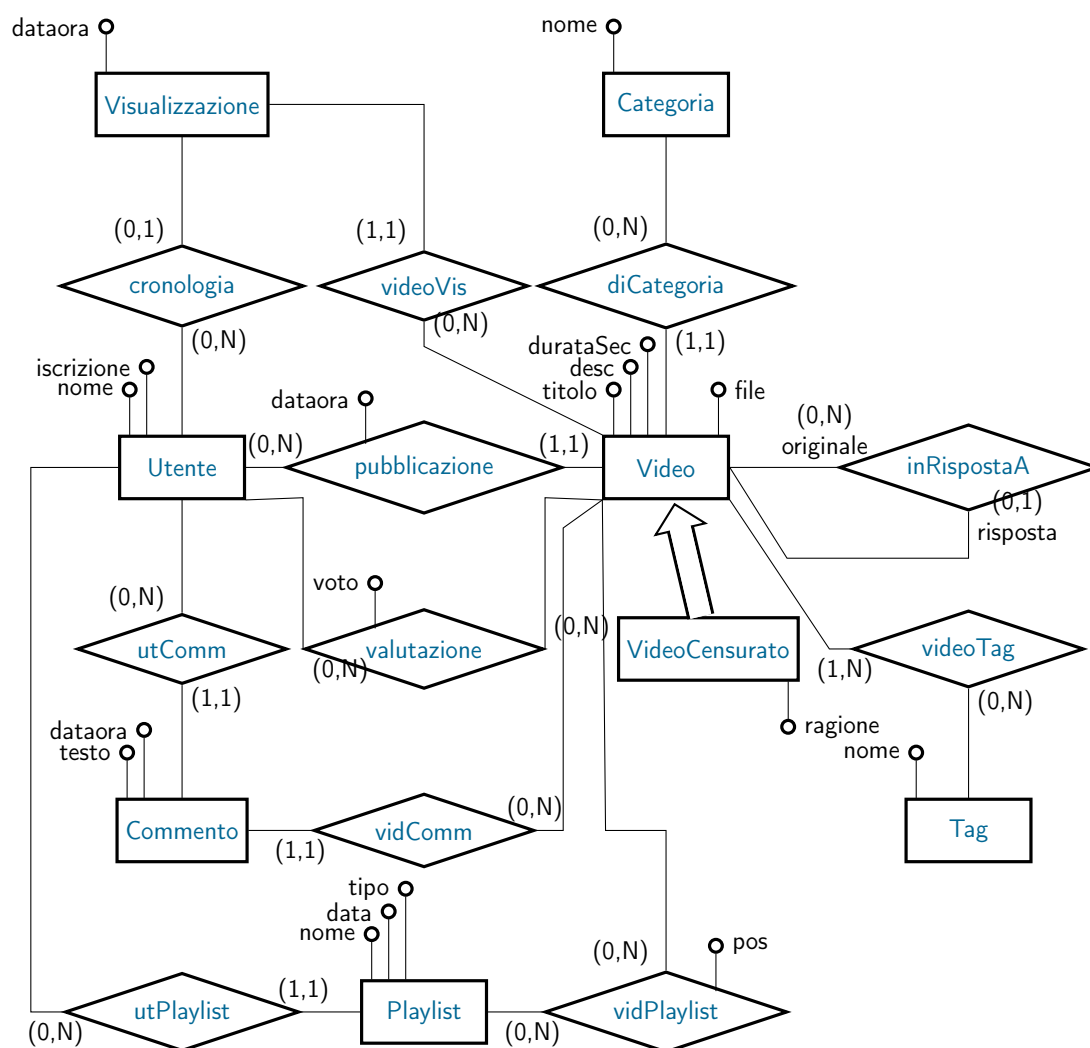
A.4

Estensione del Diagramma ER

A.4.1 Testo

Proseguire la fase di Analisi Concettuale dei requisiti estendendo il diagramma ER concettuale per l'applicazione ed il dizionario dei dati per modellare anche il seguente requisito: [Req. 8.](#)

A.4.2 Soluzione



Specifiche dei Dati

Entità **Utente**

Ogni istanza di questa entità rappresenta un utente registrato ([Req. 1.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome dell'utente (Req. 1.1.)
iscrizione	data		La data di iscrizione dell'utente al servizio (Req. 1.2.)

Entità **Video**

Ogni istanza di questa entità rappresenta un video caricato nel sistema ([Req. 2.](#))

attributo	dominio	molteplicità	descrizione
titolo	stringa		Il titolo del video (Req. 2.1.)
desc	stringa		La descrizione del video (Req. 2.2.)
durataSec	intero > 0		La durata del video in secondi (Req. 2.3.)
file	stringa		Il nome del file dove il video è memorizzato (Req. 2.4.)

Entità **VideoCensurato**

Ogni istanza di questa entità rappresenta un video censurato ([Req. 2.10.](#))

attributo	dominio	molteplicità	descrizione
ragione	stringa		La ragione della censura del video)

Entità **Visualizzazione**

Ogni istanza di questa entità rappresenta la visualizzazione di un video da parte di un utente ([Req. 5.](#))

attributo	dominio	molteplicità	descrizione
dataora	dataora		La data e l'ora della visualizzazione)

Entità **Categoria**

Ogni istanza di questa entità rappresenta una categoria ([Req. 3.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome della categoria

Entità **Tag**

Ogni istanza di questa entità rappresenta un tag ([Req. 4.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome del tag

Entità **Commento**

Ogni istanza di questa entità rappresenta un commento di un [Utente](#) ad un [Video](#) ([Req. 7.](#))

attributo	dominio	molteplicità	descrizione
testo	stringa		Il testo del commento
dataora	dataora		La data e l'ora del commento

Entità **Playlist**

Ogni istanza di questa entità rappresenta una playlist ([Req. 8.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome della playlist
data	data		La data in cui è stata creata la playlist
tipo	{pubblica, privata}		L'informazione circa il fatto che la playlist è pubblica o privata

Relationship pubblicazione

Ogni istanza di questa relationship lega un **Utente** ad uno dei **Video** che ha pubblicato (Req. 2.5.)

attributo	dominio	molteplicità	descrizione
dataora	dataora		La data e l'ora della pubblicazione

Relationship diCategoria

Ogni istanza di questa relationship lega un **Video** alla sua **Categoria** (Req. 2.6.)

Attributi: Nessuno

Relationship videoTag

Ogni istanza di questa relationship lega un **Video** ad uno dei suoi **Tag** (Req. 2.7.)

Attributi: Nessuno

Relationship inRispostaA

Ogni istanza di questa relationship lega un **Video** (ruolo 'originale') al **Video** (ruolo 'risposta') a cui risponde (Req. 2.9.)

Attributi: Nessuno

Relationship cronologia

Ogni istanza di questa relationship lega una **Visualizzazione** all'**Utente** che l'ha effettuata (Req. 5.)

Attributi: Nessuno

Relationship videoVis

Ogni istanza di questa relationship lega una **Visualizzazione** al **Video** visualizzato (Req. 5.)

Attributi: Nessuno

Relationship valutazione

Ogni istanza di questa relationship rappresenta la valutazione di video espressa da un **Utente** (Req. 6.)

attributo	dominio	molteplicità	descrizione
voto	[0, 5]		La valutazione espressa, da 0 (pessimo) a 5 (ottimo)

Relationship `utComm`

Ogni istanza di questa relationship lega un `Commento` all'`Utente` che lo ha scritto (Req. 7.1.1.)

Attributi: Nessuno

Relationship `vidComm`

Ogni istanza di questa relationship lega un `Commento` al `Video` a cui si riferisce (Req. 7.1.3.)

Attributi: Nessuno

Relationship `utPlaylist`

Ogni istanza di questa relationship lega una `Playlist` all'`Utente` che l'ha creata (Req. 8.1.)

Attributi: Nessuno

Relationship `vidPlaylist`

Ogni istanza di questa relationship lega una `Playlist` ad un `Video` in questo contenuto (Req. 8.4.)

attributo	dominio	molteplicità	descrizione
pos	intero > 0		Il numero d'ordine del video nella playlist

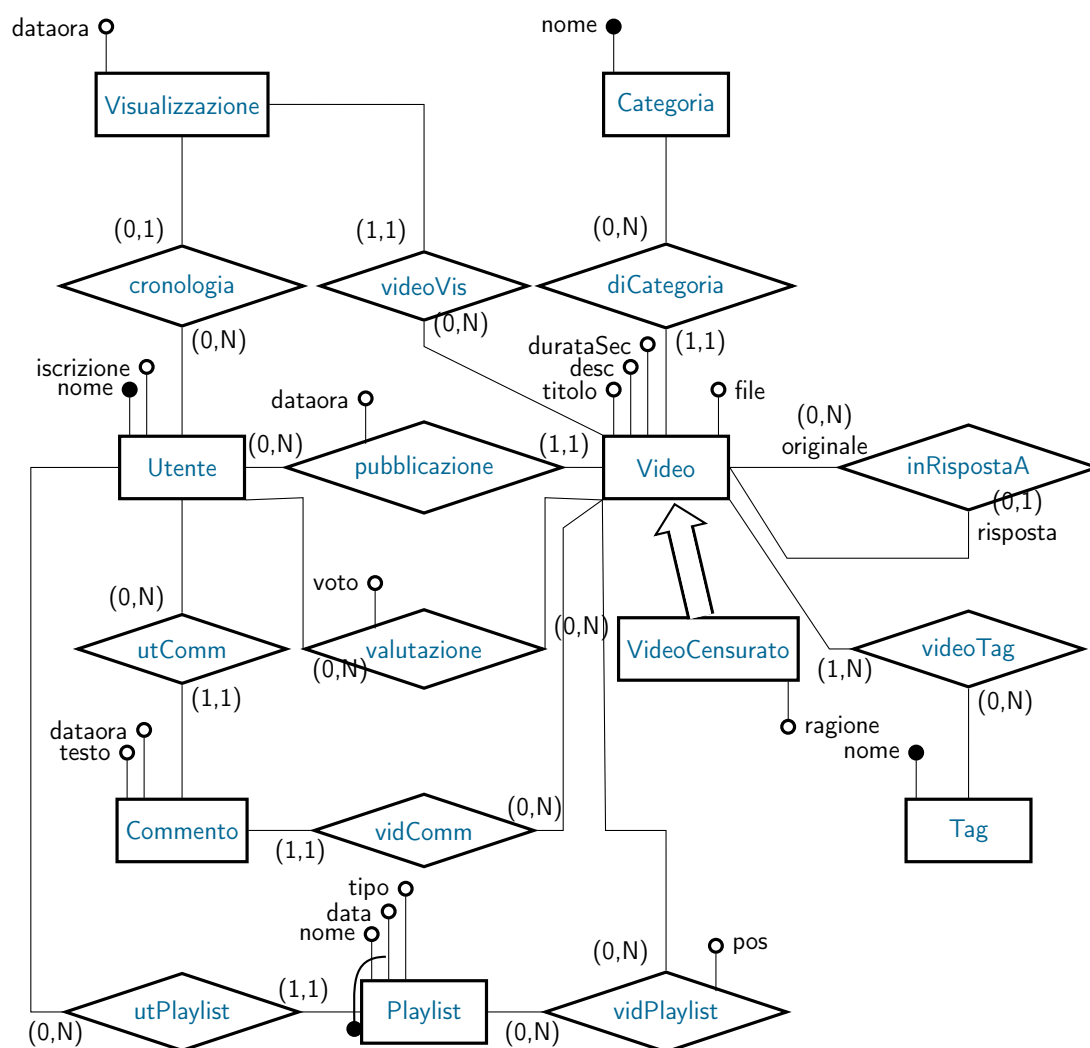
A.5

Identificazione delle Entità e Definizione Informale di Vincoli Esterni

A.5.1 Testo

Proseguire la fase di Analisi Concettuale dei requisiti estendendo il diagramma ER concettuale per l'applicazione ed il dizionario dei dati con eventuali vincoli di identificazione delle entità che provengono dai requisiti e la definizione informale di eventuali ulteriori vincoli esterni.

A.5.2 Soluzione



Specifiche dei Dati

Entità **Utente**

Ogni istanza di questa entità rappresenta un utente registrato ([Req. 1.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome dell'utente (Req. 1.1.)
iscrizione	data		La data di iscrizione dell'utente al servizio (Req. 1.2.)

Entità **Video**

Ogni istanza di questa entità rappresenta un video caricato nel sistema ([Req. 2.](#))

attributo	dominio	molteplicità	descrizione
titolo	stringa		Il titolo del video (Req. 2.1.)
desc	stringa		La descrizione del video (Req. 2.2.)
durataSec	intero > 0		La durata del video in secondi (Req. 2.3.)
file	stringa		Il nome del file dove il video è memorizzato (Req. 2.4.)

Entità **VideoCensurato**

Ogni istanza di questa entità rappresenta un video censurato ([Req. 2.10.](#))

attributo	dominio	molteplicità	descrizione
ragione	stringa		La ragione della censura del video)

Entità **Visualizzazione**

Ogni istanza di questa entità rappresenta la visualizzazione di un video da parte di un utente ([Req. 5.](#))

attributo	dominio	molteplicità	descrizione
dataora	dataora		La data e l'ora della visualizzazione)

Entità Categoria

Ogni istanza di questa entità rappresenta una categoria ([Req. 3.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome della categoria

Entità Tag

Ogni istanza di questa entità rappresenta un tag ([Req. 4.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome del tag

Entità Commento

Ogni istanza di questa entità rappresenta un commento di un [Utente](#) ad un [Video](#) ([Req. 7.](#))

attributo	dominio	molteplicità	descrizione
testo	stringa		Il testo del commento
dataora	dataora		La data e l'ora del commento

Vincoli:

[V.Commento.visualizzazione] Ogni istanza c dell'entità [Commento](#) deve soddisfare i seguenti requisiti ([Req. 7.2.](#)):

- Sia u l'[Utente](#) che ha effettuato il commento, ovvero l'istanza dell'entità [Utente](#) tale che (u, c) è istanza della relationship [utComm](#);
- Sia v il [Video](#) oggetto del commento, ovvero l'istanza dell'entità [Video](#) tale che (v, c) è istanza della relationship [vidComm](#);
- Deve esistere una istanza vis dell'entità [Visualizzazione](#) tale che:
 - (u, vis) è istanza della relationship [cronologia](#)
 - (v, vis) è istanza della relationship [videoVis](#).

Entità Playlist

Ogni istanza di questa entità rappresenta una playlist ([Req. 8.](#))

attributo	dominio	molteplicità	descrizione
nome	stringa		Il nome della playlist
data	data		La data in cui è stata creata la playlist
tipo	{pubblica, privata}		L'informazione circa il fatto che la playlist è pubblica o privata

Relationship pubblicazione

Ogni istanza di questa relationship lega un [Utente](#) ad uno dei [Video](#) che ha pubblicato ([Req. 2.5.](#))

attributo	dominio	molteplicità	descrizione
dataora	dataora		La data e l'ora della pubblicazione

Vincoli:

[V.pubblicazione.utente.registrato] L'istante di pubblicazione di un [Video](#) non può essere precedente all'istante di registrazione dell'[Utente](#) che l'ha pubblicato:

Per ogni istanza (Utente: u, Video: v) della relationship [pubblicazione](#), siano:

- ip il valore dell'attributo dataora dell'istanza (u, v)
- ir il valore dell'attributo iscrizione dell'istanza u.

Deve essere $ir \leq ip$.

Relationship diCategoria

Ogni istanza di questa relationship lega un [Video](#) alla sua [Categoria](#) ([Req. 2.6.](#))

Attributi: Nessuno

Relationship videoTag

Ogni istanza di questa relationship lega un [Video](#) ad uno dei suoi [Tag](#) ([Req. 2.7.](#))

Attributi: Nessuno

Relationship inRispostaA

Ogni istanza di questa relationship lega un **Video** (ruolo 'originale') al **Video** (ruolo 'risposta') a cui risponde (Req. 2.9.)

Attributi: Nessuno

Vincoli:

[V.inRispostaA.utente] Nessun **Utente** può pubblicare **Video** in risposta a **Video** che ha pubblicato egli stesso:

Per ogni istanza (originale: vo, risposta: vr) della relationship **inRispostaA**, siano:

- ur l'**Utente** che ha pubblicato il **Video** in risposta (vr), ovvero tale che (ur, vr) è istanza della relationship **pubblicazione**
- uo l'**Utente** che ha pubblicato il **Video** originale (vo), ovvero tale che (uo, vo) è istanza della relationship **pubblicazione**.

Deve essere $ur \neq uo$.

Relationship cronologia

Ogni istanza di questa relationship lega una **Visualizzazione** all'**Utente** che l'ha effettuata (Req. 5.)

Attributi: Nessuno

Relationship videoVis

Ogni istanza di questa relationship lega una **Visualizzazione** al **Video** visualizzato (Req. 5.)

Attributi: Nessuno

Relationship valutazione

Ogni istanza di questa relationship rappresenta la valutazione di video espressa da un **Utente** (Req. 6.)

attributo	dominio	molteplicità	descrizione
voto	[0, 5]		La valutazione espressa, da 0 (pessimo) a 5 (ottimo)

Vincoli:

[V.valutazione.visualizzazione] Per ogni istanza (Utente: u, Video: v) di questa relationship, deve esistere un'istanza vis dell'entità **Visualizzazione** tale che (Req. 6.3.):

- (v, vis) è istanza della relationship **videoVis**
- (u, vis) è istanza della relationship cronologia.

[V.valutazione.nonsuo] Per ogni istanza (**Utente** : u , **Video** : v) di questa relationship, sia u' l'**Utente** che ha pubblicato il **Video** v , ovvero l'istanza dell'entità **Utente** tale che (u', v) è istanza della relationship **pubblicazione**.

Deve essere (**Req. 6.4.**): $u \neq u'$.

Relationship **utComm**

Ogni istanza di questa relationship lega un **Commento** all'**Utente** che lo ha scritto (**Req. 7.1.1.**)

Attributi: Nessuno

Relationship **vidComm**

Ogni istanza di questa relationship lega un **Commento** al **Video** a cui si riferisce (**Req. 7.1.3.**)

Attributi: Nessuno

Relationship **utPlaylist**

Ogni istanza di questa relationship lega una **Playlist** all'**Utente** che l'ha creata (**Req. 8.1.**)

Attributi: Nessuno

Relationship **vidPlaylist**

Ogni istanza di questa relationship lega una **Playlist** ad un **Video** in questo contenuto (**Req. 8.4.**)

attributo	dominio	molteplicità	descrizione
pos	intero > 0		Il numero d'ordine del video nella playlist

Vincoli:

[V.vidPlaylist.pos] I valori delle posizioni dei **Video** di ogni **Playlist** sono tutti distinti e progressivi (**Req. 8.4.**):

- Non esistono due istanze (v, p) e (v', p) della relationship **vidPlaylist** (dove p è una istanza dell'entità **Playlist**) avente lo stesso valore per l'attributo pos;
- Per ogni istanza (v, p) della relationship **vidPlaylist** (dove v è una istanza dell'entità **Video**) avente per l'attributo pos un valore $p > 1$, deve esistere una istanza (v', p) della relationship **vidPlaylist** (dove v' è una istanza dell'entità **Video**) avente per l'attributo pos il valore $p - 1$.



A.6

Diagramma UML degli Use-Case

A.6.1 Testo

Proseguire la fase di Analisi Concettuale dei requisiti producendo un diagramma UML degli use-case.

A.6.2 Soluzione

