



**Specifiche di progetto per il corso di  
Basi di dati  
A.A. 2022 – 2023  
Corso di Laurea in Ingegneria Informatica**

Nicola Tonellotto, Francesco Pistolesi



# Indice

<b>1</b>	<b>Descrizione delle fasi di progettazione</b>	<b>3</b>
1.1	Analisi delle specifiche . . . . .	4
1.2	Progettazione concettuale . . . . .	4
1.3	Ristrutturazione del diagramma E-R . . . . .	5
1.4	Individuazione di operazioni sui dati . . . . .	5
1.5	Analisi delle prestazioni delle operazioni . . . . .	5
1.6	Progettazione logica . . . . .	6
1.7	Analisi delle dipendenze funzionali e normalizzazione . . . . .	7
1.8	Implementazione su DBMS Oracle MySQL . . . . .	7
<b>2</b>	<b>Specifiche</b>	<b>8</b>
2.1	Visione d'insieme . . . . .	8
2.2	Area contenuti . . . . .	8
2.2.1	Film . . . . .	8
2.2.2	Rating . . . . .	8
2.3	Area formati . . . . .	9
2.3.1	Tipi di formato . . . . .	9
2.3.2	Tracking dei formati e restrizioni . . . . .	9
2.4	Area clienti . . . . .	9
2.4.1	Informazioni sui clienti . . . . .	9
2.4.2	Piani di abbonamento . . . . .	10
2.4.3	Fatturazione . . . . .	10
2.4.4	Raccomandazione di contenuti . . . . .	10
2.5	Area streaming . . . . .	11
2.5.1	Content delivery network . . . . .	11
2.5.2	Server della CDN . . . . .	11
2.5.3	Caching . . . . .	12
2.6	Area analytics . . . . .	12
2.6.1	Classifiche . . . . .	13
2.6.2	Bilanciamento del carico . . . . .	13
2.6.3	Custom analytics . . . . .	13
2.7	Dimensionamento del lavoro dei gruppi in base alla loro numerosità	13



# Capitolo 1

## Descrizione delle fasi di progettazione

Si desidera progettare un database relazionale su DBMS Oracle MySQL che permetta di memorizzare i dati relativi a un sistema che gestisce *FilmSphere*, un servizio dedicato alla memorizzazione e allo streaming di contenuti video online. Oltre a ospitare i dati, il database sarà dotato di alcune operazioni e funzionalità di data analytics che dovranno essere realizzate.

Il presente capitolo ha la funzione di spiegare come dovrebbe svolgersi il lavoro e quali sono i requisiti e lo scopo di tutte le sue fasi, le tecniche che dovranno essere utilizzate, e gli artefatti che dovranno essere prodotti e mostrati in sede di verifica. Le specifiche del database da progettare e delle funzionalità da implementare sono fornite nel Capitolo 2.

Il progetto deve essere svolto in gruppi composti da **due studenti**. Solo in casi eccezionali, a fronte di disagi motivati, si può valutare l'eventualità che un gruppo sia composto da tre studenti oppure da uno; nella sezione 2.7 verranno specificate le parti che sono obbligatorie a seconda della dimensione del gruppo. I gruppi da tre studenti e da uno studente devono essere approvati dai docenti.

Il lavoro si svolgerà attraverso le seguenti fasi:

1. Analisi delle specifiche;
2. Progettazione concettuale tramite un diagramma entità-relazione;
3. Ristrutturazione del diagramma entità-relazione;
4. Individuazione di operazioni interessanti sui dati;
5. Analisi delle prestazioni delle operazioni individuate;
6. Miglioramento della performance tramite introduzione di ridondanze;
7. Traduzione dello schema concettuale nel modello logico relazionale;
8. Analisi delle dipendenze funzionali ed eventuale normalizzazione dello schema;
9. Scrittura di uno script MySQL che crei il database e lo popoli;
10. Implementazione delle funzionalità di analisi dei dati.

Come risultato finale dell'attività, oltre al **database** e allo **script** contenente il codice per la sua creazione e popolazione e per la realizzazione delle funzionalità richieste, in sede d'esame dovrà essere consegnata una **documentazione** nella quale vengono spiegate e giustificate chiaramente tutte le scelte effettuate fase per fase. Più in dettaglio, deve essere consegnata una copia digitale (in PDF) della documentazione, con pagine numerate e provvista di indice. Devono inoltre essere consegnate due copie (in PDF) del diagramma E-R, una non ristrutturata e una ristrutturata. Il diagramma E-R deve essere ordinato e chiaramente leggibile, deve essere realizzato su un foglio digitale di dimensione appropriata (tipicamente A3, A2 o formati più grandi) e il font utilizzato non dovrà essere più piccolo di 12 pt in ogni parte del diagramma. Si deve quindi usare il software di progettazione **impostando all'inizio il formato di pagina e, nel momento in cui si effettua la stampa cartacea (o in PDF), scegliere un formato carta (o di pagina digitale) dello stesso formato scelto nel software**. Per esempio, se nel software si è impostato il formato di pagina A3, quando si stampa si deve stampare in A3. Non usare opzioni "Adatta alla pagina", "Fit to page", "Fit to width", "Fit to whatever".

L'esame si svolgerà sotto forma di colloquio in cui entrambi gli studenti che compongono il gruppo sono chiamati a rispondere a domande su tutto il progetto e ad eventuali domande relative ai concetti teorici sui quali esso si basa. Qualora parti della progettazione siano svolte in autonomia dai componenti del gruppo, in sede d'esame non è accettabile che uno dei due componenti non sappia rispondere a domande relative a una parte di progettazione svolta dall'altro. Gli studenti devono inoltre dimostrare capacità critica e difendere le scelte effettuate, motivandone opportunamente le ragioni. Sarà inoltre oggetto di valutazione il livello di chiarezza espositiva. Il voto della discussione del progetto è espresso in trentesimi ed è lo stesso per entrambi gli studenti che compongono il gruppo.

## 1.1 Analisi delle specifiche

In questa fase preliminare devono essere lette e analizzate nel dettaglio le specifiche fornite nel Capitolo 2. Lo scopo dell'analisi delle specifiche è quello di capire l'utilizzo tipico del database da progettare, al fine di prevedere i dati di cui necessità e l'organizzazione migliore per il loro utilizzo.

## 1.2 Progettazione concettuale

La notazione da utilizzare per il diagramma entità-relazione è quella vista a lezione, presente sia nelle slide che nel libro di testo. Non sono accettate notazioni alternative non trattate estesamente nel corso (ad esempio UML, Crow's Foot, etc.).

Esistono vari tool per la realizzazione (più o meno) assistita di diagrammi entità-relazione. Tali tool sono reperibili sul web, taluni con licenza freeware o shareware, altri a pagamento. Alcuni esempi sono [Draw.io](#), [Microsoft Visio](#), [OmniGraffle](#) [solo macOS], [ConceptDraw](#) [solo macOS], [Adobe Illustrator](#). I tool sottolineati sono stati utilizzati in modo preferenziale dagli studenti in edizioni precedenti del corso.

Il diagramma E-R, prodotto secondo una delle strategie viste a lezione, risultante da questa fase deve contenere: entità e associazioni; attributi di entità e associazioni; identificatori primari delle entità; cardinalità delle associazioni e degli

attributi (se opzionali o multivalore); eventuali generalizzazioni, attributi composti e multivalore.

### 1.3 Ristrutturazione del diagramma E-R

Il diagramma prodotto nella fase precedente deve essere completo: la sua ristrutturazione riguarda soltanto l'eliminazione delle generalizzazioni e degli attributi composti/multivalore. Le generalizzazioni, gli attributi composti e gli attributi multivalore devono essere eliminati tramite una delle opzioni viste durante il corso. In questa fase possono anche essere effettuati partizionamenti/accorpamenti di entità e associazioni, dove si ritengano utili.

Questa fase produce come risultato un diagramma E-R ristrutturato, direttamente traducibile in uno schema nel modello logico relazionale.

### 1.4 Individuazione di operazioni sui dati

Devono essere individuate almeno 8 operazioni significative da effettuare sui dati. Queste operazioni dovranno essere implementate in linguaggio MySQL. Le operazioni individuate possono essere interrogazioni, inserimenti, modifica o cancellazioni. Con il termine "significative" si fa riferimento a operazioni che contribuiscono in maniera apprezzabile a determinare le prestazioni del database durante il normale carico applicativo<sup>1</sup>. Queste operazioni possono essere operazioni semplici ma molto frequenti, oppure operazioni più complesse di analisi dei dati che, anche se eseguite più di rado, possono appesantire considerevolmente il carico applicativo.

### 1.5 Analisi delle prestazioni delle operazioni

Nella documentazione che sarà oggetto di discussione in sede d'esame, per ogni operazione significativa individuata al punto precedente, devono essere riportati nell'ordine: una descrizione chiara dell'operazione; l'input (ciò che è noto a priori ed è dato in ingresso all'operazione); l'output (ciò che l'operazione produce); la porzione del diagramma E-R interessata dall'operazione; la porzione di tavola dei volumi (le righe) con i volumi delle entità e le relazioni coinvolte; la tavola degli accessi.

Per analizzare le prestazioni delle operazioni la tavola dei volumi dovrà essere compilata stimando il numero di occorrenze per ciascuna entità o relazione. Dopodiché, deve essere data una stima sulla frequenza giornaliera dell'esecuzione di ciascuna operazione. Affinché l'analisi sia sufficientemente indicativa, le stime devono essere fatte con giudizio. La documentazione dovrà contenere la tavola dei volumi e le assunzioni sulla frequenza delle operazioni.

Per ogni operazione scelta nel Paragrafo 1.4, la tavola degli accessi conterrà tante righe quanti sono i concetti (entità e relazioni) attraversati dall'operazione. Ogni riga della tavola degli accessi deve contenere, nell'ordine: un identificatore incrementale, il nome del concetto coinvolto, il tipo di concetto (entità o relazione), il tipo di accesso (lettura o scrittura), il numero di accessi necessari, la descrizione

---

<sup>1</sup>Solitamente, per la legge di Pareto, il 20% delle operazioni è responsabile dell'80% del carico.

della motivazione per la quale si effettuano gli accessi, cioè quale attributo (o attributi) si sta leggendo/scrivendo.

Dall'analisi delle prestazioni di ciascuna operazione significativa può emergere che alcune potrebbero trarre beneficio dall'introduzione di ridondanze. Ogniquale volta si introduce una ridondanza per una operazione, per tale operazione deve essere compilata anche la tavola degli accessi che mostra il numero di operazioni elementari<sup>2</sup> da eseguire in presenza della ridondanza. Deve infine essere presa una decisione sul mantenere o no la ridondanza introdotta, in base al risparmio di accessi che essa comporta. Si faccia attenzione che, scelta un'operazione, una ridondanza ne comporta naturalmente un alleggerimento del carico, ma, d'altra parte, la ridondanza deve essere mantenuta aggiornata per essere utilizzata. Più in dettaglio, una ridondanza richiederà un'operazione di aggiornamento (o refresh), ogniqualvolta viene eseguita una modifica (update, insert o delete) della tabella sulla quale la ridondanza è basata.

Per decidere se mantenere o meno una ridondanza, va studiata anche l'operazione di aggiornamento, la sua frequenza, la sua modalità (immediate, deferred, on demand) e la sua complessità in termini di operazioni elementari. Ciò permette di calcolare il rapporto costo-beneficio, dove il beneficio è il risparmio di operazioni elementari che la ridondanza comporta nell'esecuzione di un'operazione significativa (quelle al Paragrafo 1.4), mentre il costo è dovuto alle operazioni elementari necessarie per mantenere coerente la ridondanza. In altre parole, il carico computazionale che si introduce per mantenere aggiornata la ridondanza deve essere giustificato dal beneficio che la ridondanza produce. Per le operazioni significative che coinvolgono ridondanze, la documentazione di progetto deve contenere l'analisi costo-beneficio descritta sotto in dettaglio.

Scelta un'operazione significativa target  $T$ , avente frequenza giornaliera  $f^T$ , nella tavola degli accessi si trova il numero di operazioni elementari  $o^T$  necessari all'esecuzione dell'operazione  $T$ . Il numero di operazioni elementari giornaliere è  $n^T = f^T \cdot o^T$ . A seguito dell'introduzione della ridondanza, il numero di operazioni elementari diventerà  $o_{RID}^T$  (dove  $o_{RID}^T < o^T$ ) e il numero di operazioni giornaliere sarà  $n_{RID}^T = f^T \cdot o_{RID}^T$ . Quindi il numero di operazioni elementari risparmiate per eseguire  $T$  sarà:  $\Delta = n^T - n_{RID}^T$ .

L'operazione  $A$  che aggiorna la ridondanza avrà una frequenza giornaliera  $g^A$  e richiederà un numero di operazioni elementari  $o^A$  per l'aggiornamento e nella giornata avremo  $n^A = g^A \cdot o^A$ . La ridondanza è conveniente se il numero di operazioni elementari effettuate in presenza di ridondanza ( $n_{RID}^T + n^A$ ) è inferiore al numero effettuato da  $T$  in assenza di ridondanza, cioè se  $n^A < \Delta$ .

Nella versione finale del database, è richiesta la presenza di almeno due ridondanze. Fra le otto operazioni significative del Paragrafo 1.4 ci devono essere un'operazione di lettura e una di scrittura che impattino con essa.

## 1.6 Progettazione logica

Il diagramma entità-relazione ristrutturato deve essere tradotto nel modello logico relazionale, producendo così lo schema del database come sarebbe prodotto dall'algoritmo di traduzione automatica, compresi tutti i vincoli di integrità referenziale necessari al suo corretto impiego. La traduzione del diagramma E-R ristrutturato

<sup>2</sup>In questo ambito, un'operazione elementare è un accesso alla memoria. L'accesso in lettura comporta una operazione elementare; l'accesso in scrittura due operazioni elementari



nello schema del database non può contenere scelte progettuali, ma è solo il risultato dell'algoritmo di traduzione.

Eventuali vincoli di integrità generici devono essere implementati mediante check o trigger MySQL. È richiesta la presenza di almeno 2 vincoli di integrità generici.

## **1.7 Analisi delle dipendenze funzionali e normalizzazione**

Per ciascuna relazione (tabella) individuata, devono essere ricercate tutte le dipendenze funzionali non banali. È richiesto che il database progettato sia in forma normale di Boyce-Codd. Qualora la base di dati non lo sia, occorre effettuare una opportuna decomposizione delle relazioni non in forma normale di Boyce-Codd.

Il database potrà contenere parti non normalizzate solo a causa della presenza di ridondanze appositamente introdotte per aumentare le prestazioni. Tali ridondanze devono essere chiaramente indicate nel diagramma E-R, per esempio usando un colore dedicato.

## **1.8 Implementazione su DBMS Oracle MySQL**

Deve essere realizzato uno script MySQL per creare il database e popolare ogni sua tabella con un numero di record sufficienti a eseguire le funzionalità definite e poterne mostrare output sensati nella documentazione e durante la prova orale. Durante l'esame può infatti essere richiesta l'esecuzione di una o più funzionalità e sarà richiesto ai componenti del gruppo di spiegare il codice che le implementa e descrivere l'output. Potranno essere fatte domande relative alle modalità di implementazione.

Lo script deve contenere tutti i vincoli di integrità referenziale, i trigger che gestiscono i principali vincoli di integrità generici e business rule, e gli event e le stored procedure che realizzano le funzionalità di data analytics e di back-end descritte nei vari paragrafi del Capitolo 2.

## Capitolo 2

# Specifiche

### 2.1 Visione d'insieme

Il database che si desidera progettare ha lo scopo di memorizzare i dati a supporto delle funzionalità del sistema informativo di *FilmSphere*, un portale dedicato alla memorizzazione e streaming di contenuti video online.

### 2.2 Area contenuti

L'area contenuti è dedicata alla memorizzazione dei video (*film*, da ora in poi) e delle loro informazioni.

#### 2.2.1 Film

Ogni film ha un'identificatore univoco ed è caratterizzato da informazioni come il titolo, la descrizione, il genere, il regista, gli attori principali, la durata (espressa in minuti, o ore e minuti), e l'anno di produzione<sup>1</sup>. Un film ha anche un paese di produzione, un elenco di lingue in cui è disponibile, e un elenco di lingue disponibili per i sottotitoli. Ogni film è inoltre disponibile in vari formati audio e video. Il formato video indica il tipo di compressione video<sup>2</sup> utilizzata per il film; il formato audio indica il tipo di compressione audio utilizzata per il film<sup>3</sup>.

#### 2.2.2 Rating

Ogni film ha anche un rating che può essere espresso in varie forme, come ad esempio un punteggio numerico o un sistema di valutazione a stelle. Il rating del film potrebbe essere ad esempio generato utilizzando un algoritmo che considera diversi fattori, come le recensioni di utenti e critici, i voti dati ad altri film dello stesso genere, la popolarità del regista e degli attori principali, e il numero e la tipologia dei premi vinti dal film e dai suoi interpreti (come i premi Oscar). Per esempio, un semplice algoritmo potrebbe consentire di assegnare un punteggio da 1 a 10 a ciascuno di questi fattori, per poi calcolare il rating complessivo del film come

---

<sup>1</sup>L'anno di produzione è uguale all'anno di uscita.

<sup>2</sup>Tipici formati video includono H.265/HEVC, VP9, AV1, eccetera.

<sup>3</sup>Tipici formati audio includono MP3, AAC, e Dolby Digital.

media dei punteggi. Un film può anche un rating per ogni utente, dipendentemente dai fattori che l'utente considera importanti nella valutazione del film. Il rating del film e i rating personalizzati sono utilizzati per consigliare film sulla base dei film più visti dall'utente (e dagli altri), e sulle recensioni scritte da utenti o critici.

In sintesi, la prima area del database è dedicata ai contenuti e deve contenere tutte le informazioni necessarie a identificare e valutare ogni film presente nel database.

## 2.3 Area formati

Questa area del database riguarda i formati in cui ogni film è disponibile.

### 2.3.1 Tipi di formato

Per ogni film, è necessario tenere traccia dei diversi formati in cui è stato codificato, come ad esempio MP4, AVI, MKV, ecc. Ogni formato è identificato da un codice ed è associato a un insieme di caratteristiche tecniche che includono la risoluzione, il bitrate, la qualità audio e video, il rapporto d'aspetto, la dimensione del file, la lunghezza del video. Un formato può essere anche associato a restrizioni geografiche<sup>4</sup>. Un formato potrebbe quindi non essere fruibile in determinate aree geografiche.

### 2.3.2 Tracking dei formati e restrizioni

Il database deve tenere traccia di eventuali modifiche ai formati esistenti, inclusi gli aggiornamenti delle specifiche tecniche e le nuove versioni dei codec.

È possibile associare uno o più formati a ogni film. Le informazioni relative ai formati possono essere utilizzate per generare raccomandazioni personalizzate in base alle preferenze dell'utente e alle specifiche del dispositivo utilizzato per la visione dei contenuti.

La relazione tra la tabella dei film e quella dei formati può essere realizzata mediante una tabella di associazione, che contiene le informazioni necessarie per associare un formato ad un film specifico. In questa tabella, potrebbe essere presente anche una colonna per la data di rilascio di ogni formato, in modo da permettere di tenere traccia delle diverse versioni di un film e dei relativi aggiornamenti.

## 2.4 Area clienti

Questa area del database riguarda le informazioni relative ai clienti e alle loro interazioni con il sistema.

### 2.4.1 Informazioni sui clienti

Per garantire un'esperienza di visualizzazione personalizzata e di alta qualità, ogni utente deve essere univocamente identificato nel database. Un utente è identificato da un codice alfanumerico ed è caratterizzato da informazioni personali, quali il nome, il cognome, l'indirizzo email, e una password per l'accesso al sistema.

---

<sup>4</sup>Gli studenti possono stabilire liberamente una suddivisione geografica basata su aree definite adottando un'opportuno livello di granularità, come ad esempio a livello di Stato. In questo caso, ogni Stato è un'area geografica.

Il database deve anche memorizzare le connessioni dei clienti e i relativi istanti di inizio e fine<sup>5</sup>. A ogni connessione è anche associato un insieme di specifiche hardware del dispositivo (notebook, tablet, eccetera) utilizzato per la fruizione dei contenuti. Infine, una connessione è associata a un indirizzo IP<sup>6</sup> che consente di geolocalizzare l'utente<sup>7</sup>. Con *geolocalizzare*, si intende risalire all'area geografica da cui l'utente si collega.

## 2.4.2 Piani di abbonamento

*FilmShpere* offre varie modalità di abbonamento per soddisfare le esigenze di ogni utente. Ogni modalità di abbonamento abilita una o più caratteristiche (*pacchetti*). Per esempio, il piano "Basic" potrebbe abilitare la sola visione di contenuti in definizione standard, mentre il piano "Premium" potrebbe includere contenuti in alta definizione, contenuti esclusivi e funzionalità aggiuntive come il download offline che consente agli utenti di scaricare un film su un dispositivo di archiviazione locale (come un computer o un dispositivo mobile) per poterlo successivamente visualizzare o ascoltare senza la necessità di una connessione a Internet.

In generale, vi sono più piani di abbonamento e ciascuno garantisce la fruizione dei contenuti fornendo sottoinsiemi di funzionalità. I nomi degli abbonamenti sono "Basic", "Premium", "Pro", "Deluxe", e "Ultimate". Inoltre, per ogni utente, è necessario tenere traccia dei contenuti visualizzati, delle volte in cui sono stati visualizzati, in modo da generare consigli basati sulle preferenze dell'utente.

## 2.4.3 Fatturazione

Il database deve contenere informazioni sugli abbonamenti degli utenti, comprensivi di tariffa mensile, durata, numero massimo di ore o di gigabyte mensili, una o più tipologie o caratterizzazioni<sup>8</sup> dei contenuti visualizzabili, le funzionalità aggiuntive e le restrizioni geografiche, o di età.

Inoltre, il database mantiene traccia delle informazioni sui pagamenti degli utenti, i dettagli della carta di credito, le fatture emesse, le relative scadenze, consentendo di identificare quelle pagate, e in che data.

## 2.4.4 Raccomandazione di contenuti

Per migliorare l'esperienza dell'utente, *FilmSphere* è dotato di un sistema di raccomandazione basato sui contenuti visualizzati dagli utenti e sulle loro preferenze. Ad esempio, un utente che ha visualizzato molti film di fantascienza, nei cast dei quali comparivano determinati attori, può ricevere consigli per altri film dello stesso tipo (o simile). Il database deve mantenere tutte le informazioni necessarie affinché ciò sia possibile.

Gli studenti devono inoltre implementare una funzionalità di raccomandazione di contenuti (e memorizzarne i dati necessari) che, dato un utente, sfrutti uno o più

<sup>5</sup>Si può usare il tipo di dato `TIMESTAMP` di MySQL.

<sup>6</sup>Nella sua rappresentazione più ricorrente, un indirizzo IP è composto da 4 numeri interi in  $\{0, \dots, 255\}$ , separati da un punto. Supporre che ogni dispositivo, anche se all'interno della stessa abitazione, sia associato a un indirizzo IP univoco.

<sup>7</sup>Per semplicità, tralasciare la possibilità di connettersi al servizio usando una *Virtual Private Network* (VPN).

<sup>8</sup>Per "caratterizzazione" si intende, ad esempio, cartoni animati ad alta definizione.

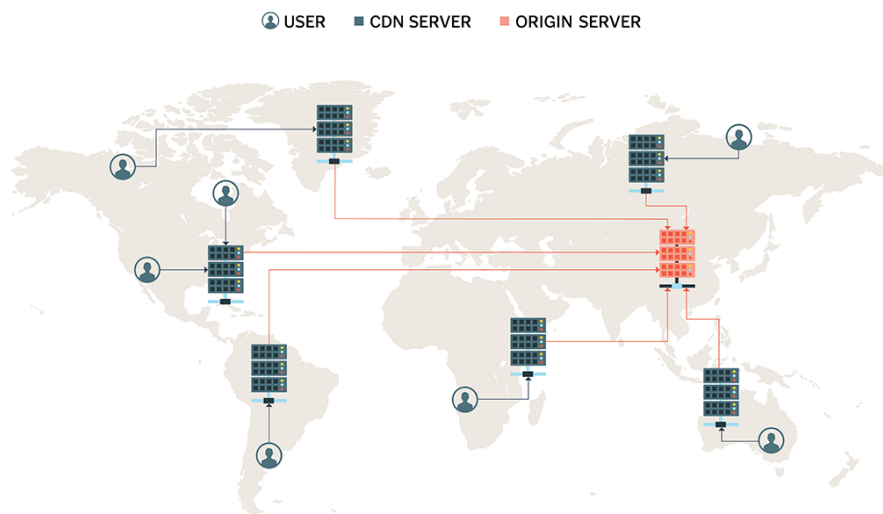


Figura 1: Tipico scenario di content delivery network.

criteri opportunamente definiti per suggerire film potenzialmente graditi al cliente, in base alla sua *"history" di visualizzazioni*.

## 2.5 Area streaming

L'ultima area del database riguarda i dettagli circa la rete di streaming dei contenuti: la *content delivery network (CDN)*.

### 2.5.1 Content delivery network

La CDN è una rete di distribuzione di contenuti costituita da server<sup>9</sup>, distribuiti geograficamente, che aiutano a ridurre i rallentamenti e gli episodi di lag durante la visualizzazione dei contenuti, accorciando virtualmente la distanza fisica tra l'utente e il server che ospita il contenuto richiesto.

Operativamente, quando un utente richiede di visualizzare un film, il sistema utilizza la CDN, la quale sfrutta l'insieme di server distribuiti in tutto il mondo che la compongono, al fine di fornire il contenuto video all'utente in modo rapido e senza interruzioni. La Figura 1 mostra uno scenario di content delivery network.

### 2.5.2 Server della CDN

Ogni server del CDN è memorizzato nel database tramite un identificatore, una posizione geografica, una larghezza di banda e una capacità massima di trasmissione. Inoltre, il database deve contenere informazioni relative al monitoraggio continuo dello stato dei server del CDN, per garantire che tutti i server possano gestire il carico di traffico dovuto alle richieste di visualizzazione dei film. Ogni server, in un

<sup>9</sup>Un server è una macchina connessa in rete il cui compito è erogare un servizio, in questo caso, l'invio di un contenuto multimediale.

dato momento, avrà quindi uno stato dipendentemente dalla quantità di dati che sta fornendo.

Quando un utente richiede di visualizzare un film al servizio, una funzionalità, da implementare, sceglie il server del CDN più vicino all'utente che ha il contenuto richiesto, per ridurre i tempi di buffering e migliorare l'esperienza di visione. Inoltre, questa funzionalità considera le informazioni sul traffico di rete e sulle prestazioni dei vari server del CDN, al fine di garantire che l'utente riceva il contenuto dal server *più opportuno* in quel momento, non solo in ottica di prossimità, ma anche in termini di bilanciamento del carico.

### 2.5.3 Caching

È possibile utilizzare diverse tecniche di caching per velocizzare la visualizzazione ed evitare il buffering. Ad esempio, i film più richiesti da un utente possono essere memorizzati in cache sui server del CDN più vicini all'utente (detti *edge server*), che costituiscono un Point of Presence (PoP) del film. Inoltre, i dati relativi alla posizione geografica dell'utente e alle preferenze di visualizzazione possono essere utilizzati per prevedere la domanda futura, e posizionare in anticipo i contenuti nei server del CDN più opportuni.

Si richiede di implementare una funzionalità che, dato un utente, preveda i contenuti di cui tale utente potrebbe richiedere la visualizzazione, permettendo così di fare caching di questi contenuti in opportuni server del CDN. Definire una metrica per misurare la probabilità che un contenuto venga richiesto da un utente. Qualsiasi metrica è corretta, purché ne venga spiegato il ragionamento sottostante all'interno della documentazione.

Durante la visione di un film da parte di un utente, il server CDN può cambiare. Questo può avvenire per diversi motivi, come ad esempio il sovraccarico del server, o il sopraggiungere di problemi di connessione. Il sistema di CDN è progettato per scegliere automaticamente il server (alternativo) più opportuno, cioè vicino all'utente e in grado di garantire una visualizzazione fluida e senza interruzioni dato il suo livello di carico. Inoltre, il sistema di CDN può utilizzare tecniche di caching per mantenere una copia del film in diversi server, in modo da poterlo fornire più velocemente e ridurre il tempo di buffering.

## 2.6 Area analytics

L'area analytics contiene funzionalità lato server (data tier) che permettono di analizzare i dati alla ricerca di informazioni utili a migliorare sia l'esperienza utente che l'efficienza della CDN.

Gli studenti devono implementare le funzionalità di analisi dei dati descritte nei seguenti paragrafi con spirito d'iniziativa. In questa fase della progettazione, sarà valutata la creatività e la capacità di proporre soluzioni che, seppur semplici, rappresentino strumenti di analisi dei dati utili nel contesto applicativo descritto in questo documento di specifica.

Nella documentazione consegnata in fase di discussione del progetto, le funzionalità implementate devono essere spiegate in dettaglio, in un paragrafo dedicato, mettendone in luce i punti di forza ed eventuali aspetti critici, anche attraverso la descrizione di scenari di esecuzione. Non è sufficiente inserire il codice commentato.

### 2.6.1 Classifiche

Questa funzionalità si occupa di stilare delle classifiche dei contenuti più visti e dei loro formati, a seconda della zona geografica, e della modalità di abbonamento.

Gli studenti devono studiare e implementare una funzionalità che costruisca le suddette classifiche.

Non è importante che la funzionalità sia complessa e articolata. Qualsiasi soluzione proposta è corretta, purché ne sia giustificato il ragionamento nella documentazione.

### 2.6.2 Bilanciamento del carico

A fronte di un'analisi dei dati della CDN e dell'uso fatto dai clienti dei film memorizzati nella CDN, questa funzionalità propone piani di allocazione dei contenuti sui server al fine di non sovraccaricare il traffico di richieste di visualizzazione dei film.

Gli studenti devono studiare e implementare una funzionalità che stimi i potenziali sovraccarichi di un server, e suggerisca spostamenti di contenuti da un server per non sovravvarcarlo, indicando un server di destinazione.

Non è importante che la funzionalità sia complessa e articolata. Qualsiasi soluzione proposta è corretta, purché ne sia giustificato il ragionamento nella documentazione.

### 2.6.3 Custom analytics

Gli studenti propongano una funzionalità analytics a piacere e spieghino nella documentazione l'importanza e l'impiego di tale funzionalità, nonché le scelte fatte per l'implementazione.

## 2.7 Dimensionamento del lavoro dei gruppi in base alla loro numerosità

Il progetto deve essere svolto da **2 studenti**. Solo in casi eccezionali, per ragioni ben motivate e previa autorizzazione dei docenti, i gruppi possono essere composti da 3 studenti, oppure da 1 solo. Il lavoro svolto deve rispettare quanto segue:

- i gruppi composti da 3 studenti svolgeranno il progetto nella sua interezza.
- i gruppi composti da 2 studenti possono non realizzare una funzionalità, a scelta, tra quelle descritte nel paragrafo 2.6, e possono non realizzare il meccanismo delle raccomandazioni dei contenuti sulla base delle preferenze degli utenti nel paragrafo 2.4.4.
- i gruppi composti da 1 studente possono non realizzare l'Area descritta nel paragrafo 2.5 e le Aree analytics 2.6.2 e 2.6.3. L'Area analytics 2.6.1 è l'unica obbligatoria.