

Catalogo Video

ELABORATO CONCLUSIVO DEL CORSO BASI DI DATI E WEB

Barcella Gabriele, Bordogna Filippo | Basi di Dati e Web | 8 Giugno 2020

# Requisiti dell’applicazione

Si deve progettare una base di dati per un sito Web dedicato ai contenuti video (film, serie televisive, saghe e documentari).

Il sito organizza una comunità di utenti, ciascuno con un proprio account.

Bisognerà quindi prevedere una pagina di login e una pagina di registrazione.

Ogni utente fornisce contenuti “ricchi” riguardo i singoli video e le serie:

* Recensioni (voto ed eventuale commento).
* Curiosità.

Il sistema descrive i video in modo esteso, descrivendo le persone che partecipano alla realizzazione di un contenuto (attori, produttori, registi) e il legame eventuale con altri contenuti (appartenenza ad una saga o serie).

Deve essere possibile consultare una pagina Web per ogni persona, con l’indicazione di tutti i ruoli (attore, regista, produttore) e i corrispondenti video.

Anche per i personaggi interpretati deve essere possibile visualizzare una scheda completa che descrive tutte le occasioni in cui un personaggio è presente in un video.

I contenuti “ricchi” caricati dagli utenti devono essere verificati da un dipendente (utente admin) prima di essere visibili sul sito.

Bisogna tenere traccia delle autorizzazioni sui contenuti generate dai dipendenti.

Per poter individuare situazioni in cui lo stesso utente sfrutta identificativi per manipolare il sistema, bisogna tenere traccia degli ultimi 10 accessi di ciascun utente, con la memorizzazione dell’indirizzo di rete e dell’inizio e durata della sessione di navigazione.

Occorre implementare una barra di ricerca tramite la quale è possibile cercare film, documentari, serie, saghe, persone che partecipano alla realizzazione di film e personaggi.

Tramite un menù deve essere possibile selezionare una categoria e vedere tutti i contenuti appartenenti ad essa.

Una volta loggato l’utente deve avere la possibilità di visualizzare il suo profilo, dove può modificare la password, e tutte le sue recensioni e curiosità, oltre ad effettuare il logout.

# Progetto concettuale (ER) e logico

Essendo gli schemi particolarmente complicanti e difficilmente rappresentabili in un spazio così ristretto come un foglio Word abbiamo deciso di rappresentarlo nel file <SchemaER-Logico-Finale.jpg>

# Popolazione della base di dati d’esempio

Di seguito sono mostrate le istruzioni per la creazione delle tabelle ed alcuni esempi (2 tuple) di popolazione delle stesse in linguaggio SQL:

CREATE TABLE `accessi` (

`id` int(11) NOT NULL,

`indirizzoIP` varchar(18) NOT NULL,

`dataOra` timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

`durata` int(11) DEFAULT NULL,

`idUtente` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `accessi` (`id`, `indirizzoIP`, `dataOra`, `durata`, `idUtente`) VALUES

(1, '::1', '2020-05-26 22:20:11', 83, 3),

(2, '::1', '2020-05-26 22:29:23', 906, 4);

CREATE TABLE `attorivideo` (

`idVideo` int(11) NOT NULL,

`idPersona` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf16;

INSERT INTO `attorivideo` (`idVideo`, `idPersona`) VALUES

(1, 1),

(1, 10);

CREATE TABLE `comparizioni` (

`idPersonaggio` int(11) NOT NULL,

`idVideo` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `comparizioni` (`idPersonaggio`, `idVideo`) VALUES

(1, 1),

(1, 3);

CREATE TABLE `curiositaserie` (

`id` int(11) NOT NULL,

`idSerie` int(11) NOT NULL,

`idUtente` int(11) NOT NULL,

`idAdmin` int(11) DEFAULT NULL,

`testo` varchar(255) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `curiositavideo` (

`id` int(11) NOT NULL,

`idVideo` int(11) NOT NULL,

`idUtente` int(11) NOT NULL,

`idAdmin` int(11) DEFAULT NULL,

`testo` varchar(255) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `curiositavideo` (`id`, `idVideo`, `idUtente`, `idAdmin`, `testo`) VALUES

(6, 1, 3, NULL, 'prova prova'),

(7, 19, 7, 3, 'Rispetto al libro manca una scena in cui Katniss rischia di morire di sete');

CREATE TABLE `generi` (

`id` int(11) NOT NULL,

`tipo` varchar(50) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf16;

INSERT INTO `generi` (`id`, `tipo`) VALUES

(1, 'Animazone'),

(2, 'Avventura');

CREATE TABLE `generivideo` (

`id` int(11) NOT NULL,

`idVideo` int(11) NOT NULL,

`idGenere` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf16;

INSERT INTO `generivideo` (`id`, `idVideo`, `idGenere`) VALUES

(31, 1, 2),

(1, 1, 3);

CREATE TABLE `interpretazioni` (

`idPersona` int(11) NOT NULL,

`idPersonaggio` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `interpretazioni` (`idPersona`, `idPersonaggio`) VALUES

(1, 1),

(2, 2);

CREATE TABLE `personaggi` (

`id` int(11) NOT NULL,

`nome` varchar(50) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `personaggi` (`id`, `nome`) VALUES

(10, 'Ant-Man'),

(17, 'Boaz Rein-Buskila');

CREATE TABLE `persone` (

`id` int(11) NOT NULL,

`nome` varchar(50) NOT NULL,

`cognome` varchar(50) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `persone` (`id`, `nome`, `cognome`) VALUES

(24, ' Alex ', 'Rodrigo'),

(61, 'Alain', 'Resnais');

CREATE TABLE `produttorivideo` (

`idVideo` int(11) NOT NULL,

`idPersona` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf16;

INSERT INTO `produttorivideo` (`idVideo`, `idPersona`) VALUES

(1, 10),

(3, 10);

CREATE TABLE `recensioniserie` (

`idSerie` int(11) NOT NULL,

`idUtente` int(11) NOT NULL,

`voto` int(11) NOT NULL,

`testo` varchar(255) DEFAULT NULL,

`idAdmin` int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `recensioniserie` (`idSerie`, `idUtente`, `voto`, `testo`, `idAdmin`) VALUES

(1, 1, 5, 'Non un granchè', 6),

(1, 7, 8, 'Molto carina. Mi sono sfuggiti alcuni passaggi', 3);

CREATE TABLE `recensionivideo` (

`idVideo` int(11) NOT NULL,

`idUtente` int(11) NOT NULL,

`voto` int(11) NOT NULL,

`testo` varchar(255) DEFAULT NULL,

`idAdmin` int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `recensionivideo` (`idVideo`, `idUtente`, `voto`, `testo`, `idAdmin`) VALUES

(1, 1, 10, 'IRON MAN SEI IL MIO EROE! ', 5),

1,5,9, 'Mi è piaciuto molto, ottimo da vedere mentre si mangia un bel piatto di polenta', 3);

CREATE TABLE `registivideo` (

`idVideo` int(11) NOT NULL,

`idPersona` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf16;

INSERT INTO `registivideo` (`idVideo`, `idPersona`) VALUES

(1, 10),

(2, 13);

CREATE TABLE `saghe` (

`id` int(11) NOT NULL,

`nome` varchar(50) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `saghe` (`id`, `nome`) VALUES

(3, 'Hunger Games'),

(1, 'Marvel Cinematic Universe');

CREATE TABLE `serie` (

`id` int(11) NOT NULL,

`nome` varchar(50) NOT NULL,

`sinossi` varchar(500) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `serie` (`id`, `nome`, `sinossi`) VALUES

(1, 'La casa di Carta', 'La storia narra gli sviluppi di una rapina estremamente ambiziosa e originale: irrompere nella Fábrica Nacional de Moneda y Timbre, a Madrid, far stampare migliaia di milioni di banconote e scappare con il bottino.'),

(2, 'Person of Interest', 'In seguito agli attentati che colpirono gli Stati Uniti nel 2001, il misterioso genio dell\'informatica miliardario Harold Finch ha costruito La Macchina, un\'intelligenza artificiale (IA), per un progetto segreto antiterroristico dell\'amministrazione americana chiamato Northern Lights.');

CREATE TABLE `utenti` (

`id` int(11) NOT NULL,

`username` varchar(50) NOT NULL,

`email` varchar(90) NOT NULL,

`password` varchar(32) NOT NULL,

`admin` tinyint(1) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `utenti` (`id`, `username`, `email`, `password`, `admin`) VALUES

(1, 'username', 'email@email.com', '5f4dcc3b5aa765d61d8327deb882cf99', 0),

(3, 'grande capo', 'admin@admin.it', '21232f297a57a5a743894a0e4a801fc3', 1);

CREATE TABLE `video` (

`id` int(11) NOT NULL,

`nome` varchar(50) NOT NULL,

`durata` int(11) NOT NULL,

`idSaga` int(11) DEFAULT NULL,

`idSerie` int(11) DEFAULT NULL,

`numero` int(11) DEFAULT NULL,

`stagione` int(11) DEFAULT NULL,

`selettore` int(11) NOT NULL,

`sinossi` varchar(500) DEFAULT NULL,

`annoUscita` year(4) NOT NULL,

`nazionalita` varchar(3) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `video` (`id`, `nome`, `durata`, `idSaga`, `idSerie`, `numero`, `stagione`, `selettore`, `sinossi`, `annoUscita`, `nazionalita`) VALUES

(1, 'Iron Man', 126, 1, NULL, 1, NULL, 1, 'Dopo essere sopravvissuto ad un attacco inaspettato in territorio nemico, l\'industriale Tony Stark costruisce un\'armatura ad alta tecnologia e giura di proteggere il mondo nei panni di Iron Man.', 2008, 'USA'),

(2, 'L\'incredibile Hulk', 135, 1, NULL, 2, NULL, 1, 'Bruce Banner era uno scienziato, ma un\'esposizione accidentale ai raggi gamma ha provocato una mutazione genetica e sconvolto la sua esistenza. Ogni qualvolta le emozioni lo assalgono, Bruce si trasforma in Hulk, mostro verde dalla forza smisurata.', 2008, 'USA');

Di seguito sono mostrati i vincoli sulle chiavi primarie e sui campi univoci:

ALTER TABLE `accessi`

ADD PRIMARY KEY (`id`);

ALTER TABLE `attorivideo`

ADD PRIMARY KEY (`idVideo`,`idPersona`);

ALTER TABLE `comparizioni`

ADD PRIMARY KEY (`idPersonaggio`,`idVideo`);

ALTER TABLE `curiositaserie`

ADD PRIMARY KEY (`id`);

ALTER TABLE `curiositavideo`

ADD PRIMARY KEY (`id);

ALTER TABLE `generi`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `Tipo` (`tipo`),

ADD UNIQUE KEY `tipo\_2` (`tipo`);

ALTER TABLE `generivideo`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `idVideo\_2` (`idVideo`,`idGenere`);

ALTER TABLE `interpretazioni`

ADD PRIMARY KEY (`idPersona`,`idPersonaggio`);

ALTER TABLE `personaggi`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `nome` (`nome`);

ALTER TABLE `persone`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `nome` (`nome`,`cognome`);

ALTER TABLE `produttorivideo`

ADD PRIMARY KEY (`idVideo`,`idPersona`);

ALTER TABLE `recensioniserie`

ADD PRIMARY KEY (`idSerie`,`idUtente`);

ALTER TABLE `recensionivideo`

ADD PRIMARY KEY (`idVideo`,`idUtente`);

ALTER TABLE `registivideo`

ADD PRIMARY KEY (`idVideo`,`idPersona`),

ADD KEY `idRegista` (`idPersona`);

ALTER TABLE `saghe`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `nome` (`nome`);

ALTER TABLE `serie`

ADD PRIMARY KEY (`id`);

ALTER TABLE `utenti`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `username` (`username`),

ADD UNIQUE KEY `email` (`email`);

ALTER TABLE `video`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `idSerie\_2` (`idSerie`,`numero`,`stagione`),

ADD UNIQUE KEY `idSaga\_2` (`idSaga`,`numero`);

Di seguito sono mostrati i vincoli sulle chiavi esterne:

ALTER TABLE `accessi`

ADD CONSTRAINT `accessi\_ibfk\_1` FOREIGN KEY (`idUtente`) REFERENCES `utenti` (`id`);

ALTER TABLE `attorivideo`

ADD CONSTRAINT `attorivideo\_ibfk\_1` FOREIGN KEY (`idPersona`) REFERENCES `persone` (`id`),

ADD CONSTRAINT `attorivideo\_ibfk\_2` FOREIGN KEY (`idVideo`) REFERENCES `video` (`id`);

ALTER TABLE `comparizioni`

ADD CONSTRAINT `comparizioni\_ibfk\_1` FOREIGN KEY (`idPersonaggio`) REFERENCES `personaggi` (`id`),

ADD CONSTRAINT `comparizioni\_ibfk\_2` FOREIGN KEY (`idVideo`) REFERENCES `video` (`id`);

ALTER TABLE `curiositaserie`

ADD CONSTRAINT `curiositaserie\_ibfk\_1` FOREIGN KEY (`idSerie`) REFERENCES `serie` (`id`),

ADD CONSTRAINT `curiositaserie\_ibfk\_2` FOREIGN KEY (`idUtente`) REFERENCES `utenti` (`id`),

ADD CONSTRAINT `curiositaserie\_ibfk\_3` FOREIGN KEY (`idAdmin`) REFERENCES `utenti` (`id`);

ALTER TABLE `curiositavideo`

ADD CONSTRAINT `curiositavideo\_ibfk\_1` FOREIGN KEY (`idVideo`) REFERENCES `video` (`id`),

ADD CONSTRAINT `curiositavideo\_ibfk\_2` FOREIGN KEY (`idUtente`) REFERENCES `utenti` (`id`),

ADD CONSTRAINT `curiositavideo\_ibfk\_3` FOREIGN KEY (`idAdmin`) REFERENCES `utenti` (`id`);

ALTER TABLE `generivideo`

ADD CONSTRAINT `generivideo\_ibfk\_1` FOREIGN KEY (`idVideo`) REFERENCES `video` (`id`),

ADD CONSTRAINT `generivideo\_ibfk\_2` FOREIGN KEY (`idVideo`) REFERENCES `video` (`id`),

ADD CONSTRAINT `generivideo\_ibfk\_3` FOREIGN KEY (`idGenere`) REFERENCES `generi` (`id`);

ALTER TABLE `interpretazioni`

ADD CONSTRAINT `interpretazioni\_ibfk\_1` FOREIGN KEY (`idPersona`) REFERENCES `persone` (`id`),

ADD CONSTRAINT `interpretazioni\_ibfk\_2` FOREIGN KEY (`idPersonaggio`) REFERENCES `personaggi` (`id`);

ALTER TABLE `produttorivideo`

ADD CONSTRAINT `produttorivideo\_ibfk\_1` FOREIGN KEY (`idPersona`) REFERENCES `persone` (`id`),

ADD CONSTRAINT `produttorivideo\_ibfk\_2` FOREIGN KEY (`idVideo`) REFERENCES `video` (`id`),

ADD CONSTRAINT `produttorivideo\_ibfk\_3` FOREIGN KEY (`idVideo`) REFERENCES `video` (`id`);

ALTER TABLE `recensioniserie`

ADD CONSTRAINT `recensioniserie\_ibfk\_1` FOREIGN KEY (`idSerie`) REFERENCES `serie` (`id`),

ADD CONSTRAINT `recensioniserie\_ibfk\_2` FOREIGN KEY (`idUtente`) REFERENCES `utenti` (`id`),

ADD CONSTRAINT `recensioniserie\_ibfk\_3` FOREIGN KEY (`idAdmin`) REFERENCES `utenti` (`id`);

ALTER TABLE `recensionivideo`

ADD CONSTRAINT `recensionivideo\_ibfk\_1` FOREIGN KEY (`idVideo`) REFERENCES `video` (`id`),

ADD CONSTRAINT `recensionivideo\_ibfk\_2` FOREIGN KEY (`idUtente`) REFERENCES `utenti` (`id`),

ADD CONSTRAINT `recensionivideo\_ibfk\_3` FOREIGN KEY (`idAdmin`) REFERENCES `utenti` (`id`);

ALTER TABLE `registivideo`

ADD CONSTRAINT `registivideo\_ibfk\_1` FOREIGN KEY (`idPersona`) REFERENCES `persone` (`id`),

ADD CONSTRAINT `registivideo\_ibfk\_2` FOREIGN KEY (`idVideo`) REFERENCES `video` (`id`);

ALTER TABLE `video`

ADD CONSTRAINT `video\_ibfk\_1` FOREIGN KEY (`idSaga`) REFERENCES `saghe` (`id`),

ADD CONSTRAINT `video\_ibfk\_2` FOREIGN KEY (`idSerie`) REFERENCES `serie` (`id`);

Nel file <catalogo.sql> sono presenti tutte le query per la creazione e popolazione del DB. Tra le cose tralasciate è presente la definizione degli indici.