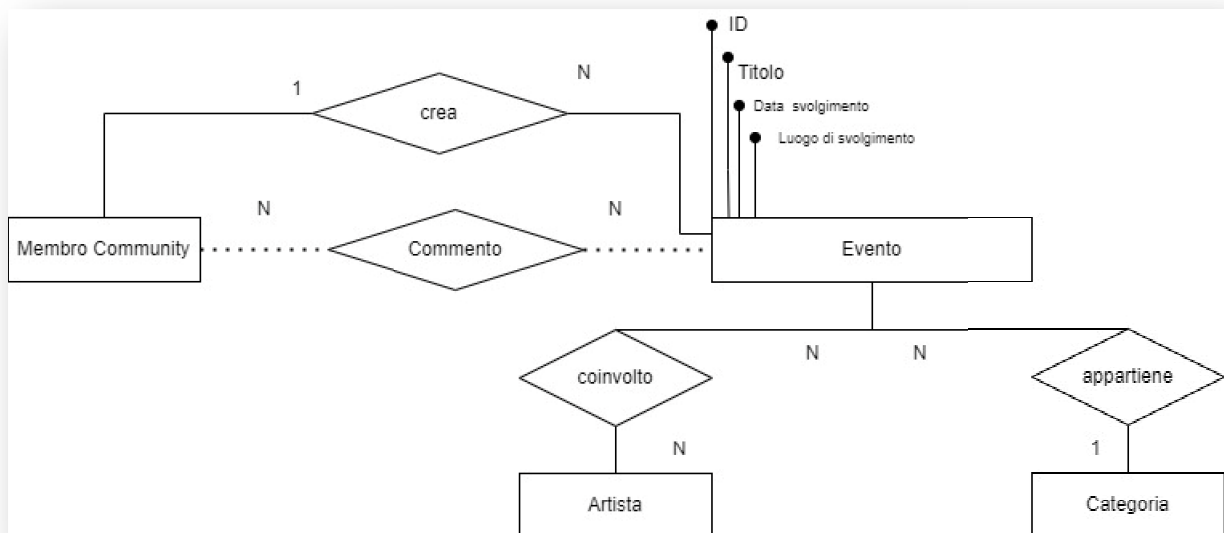
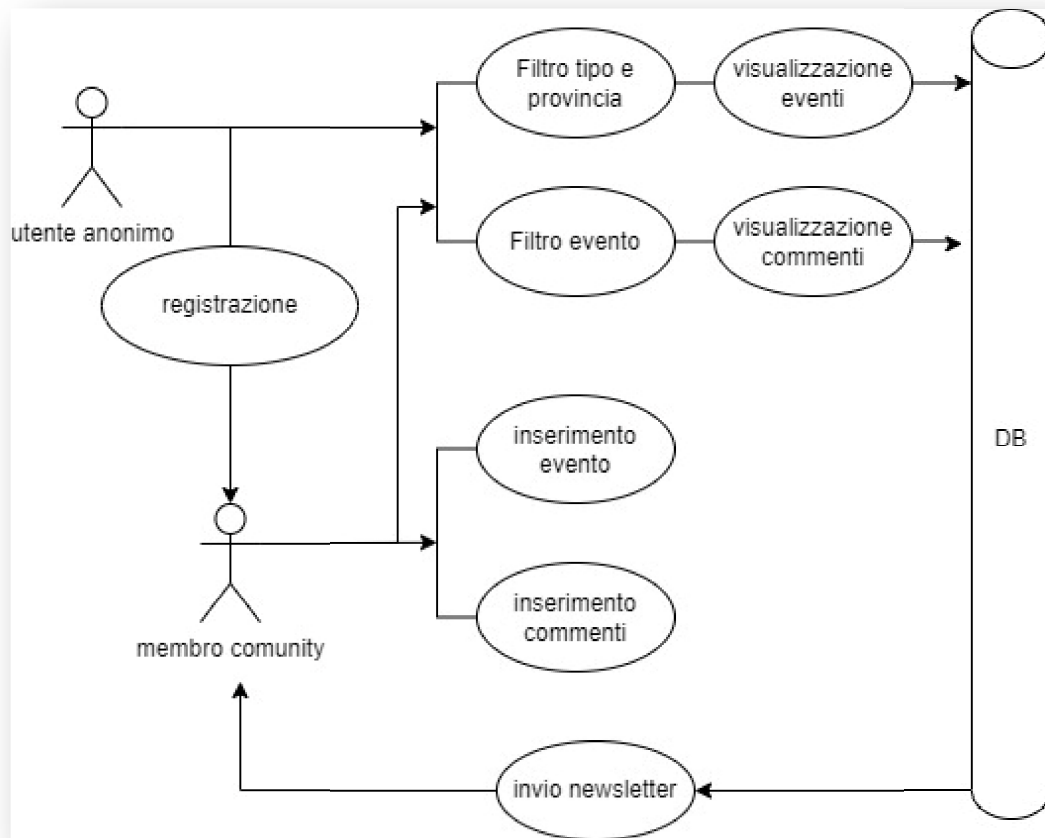


Lavoro svolto dall'alunno Domenico Auriemma VC inf as 2021/2022

Soluzione II prova scritta Esame di Stato 2015

Si vuole realizzare una web community per condividere dati e commenti relativi a eventi dal vivo di diverse categorie, ad esempio concerti, spettacoli teatrali, balletti, ecc. che si svolgono in Italia. Gli eventi vengono inseriti sul sistema direttamente dai membri stessi della community, che si registrano sul sito fornendo un nickname, nome, cognome, indirizzo di e-mail e scegliendo una o più categorie di eventi a cui sono interessati. Ogni membro iscritto riceve periodicamente per posta elettronica una newsletter, emessa automaticamente dal sistema, che riporta gli eventi delle categorie da lui scelte, che si svolgeranno nella settimana seguente nel territorio provinciale dell'utente. I membri registrati possono interagire con la community sia inserendo i dati di un nuovo evento, per il quale occorre specificare categoria, luogo di svolgimento, data, titolo dell'evento e artisti coinvolti, sia scrivendo un post con un commento ed un voto (da 1 a 5) su un evento. Il sito della community offre a tutti, sia membri registrati sia utenti anonimi, la consultazione dei dati on line, tra cui: visualizzazione degli eventi di un certo tipo in ordine cronologico, con possibilità di filtro per– territorio di una specifica provincia visualizzazione di tutti i commenti e voti relativi ad un evento.– Il candidato, fatte le opportune ipotesi aggiuntive, sviluppi

1. un'analisi della realtà di riferimento individuando le possibili soluzioni e scelga quella che a suo motivato giudizio è la più idonea a rispondere alle specifiche indicate
2. uno schema concettuale della base di dati
3. uno schema logico della base di dati
4. la definizione in linguaggio SQL di un sottoinsieme delle relazioni della base di dati in cui siano presenti alcune di quelle che contengono vincoli di integrità referenziale e/o vincoli di dominio, laddove presenti
5. le seguenti interrogazioni espresse in linguaggio SQL: a. elenco degli eventi già svolti, in ordine alfabetico di provincia b. elenco dei membri che non hanno mai inserito un commento c. per ogni evento il voto medio ottenuto in ordine di categoria e titolo d. i dati dell'utente che ha registrato il maggior numero di eventi 6. il progetto della pagina dell'interfaccia WEB che permetta ad un utente registrato di svolgere le operazioni specificate 7. la codifica in un linguaggio a scelta di un segmento significativo dell'applicazione Web che consente l'interazione con la base di dati.



```
CREATE TABLE `artisti` ( `id_artista` VARCHAR(10) NOT NULL , `nome` VARCHAR(80) NOT NULL ,  
`cognome` VARCHAR(80) NOT NULL , `num_telefono` VARCHAR(10) NOT NULL , PRIMARY KEY (`id_artista`))  
ENGINE = InnoDB;
```

```
CREATE TABLE `categorie` ( `id_categoria` VARCHAR(20) NOT NULL , `descrizione` VARCHAR(500) NOT  
NULL ,  
PRIMARY KEY (`id_categoria`)) ENGINE = InnoDB;
```

```
CREATE TABLE `membri_community` ( `id_utente` VARCHAR(20) NOT NULL , `nome` VARCHAR(80) NOT  
NULL ,  
`cognome` VARCHAR(80) NOT NULL , `email` VARCHAR(80) NOT NULL , PRIMARY KEY (`id_utente`)) ENGINE  
= InnoDB;
```

```
CREATE TABLE `eventi` ( `id_evento` VARCHAR(20) NOT NULL , `titolo` VARCHAR(80) NOT NULL ,  
`data_evento` DATE NOT NULL , `luogo_evento` ENUM('caserta','napoli','salerno','avellino','benevento')  
NOT NULL ,  
`id_categoria` VARCHAR(20) NOT NULL , `id_utente` VARCHAR(20) NOT NULL, PRIMARY KEY (`id_evento`),  
FOREIGN KEY (`id_categoria`) REFERENCES categorie(`id_categoria`),  
FOREIGN KEY (`id_utente`) REFERENCES membri_community(`id_utente`)) ENGINE = InnoDB;
```

```
CREATE TABLE eventi_artisti(id_eventovarchar(20), id_artistavarchar(20), FOREIGN KEY (id_evento)  
REFERENCES eventi(id_evento), FOREIGN KEY (id_artista) REFERENCES artisti(id_artista)) ENGINE = INNODB;
```

```
CREATE TABLE `commenti` ( `id_utente` VARCHAR(20) NOT NULL , `id_evento` VARCHAR(20) NOT NULL ,  
`commento` VARCHAR(500) NOT NULL , `rating` ENUM('1','2','3','4','5') NOT NULL, FOREIGN KEY (id_utente)  
REFERENCES membri_community(id_utente), FOREIGN KEY (id_evento) REFERENCES eventi(id_evento) )  
ENGINE = InnoDB;
```

```
INSERT INTO `artisti` (`id_artista`, `nome`, `cognome`, `num_telefono`) VALUES
('a1', 'michele', 'rech', '1111111111'), ('a2', 'ariana', 'grande', '2222222222'),
('a3', 'jeff', 'lemire', '3333333333'), ('a4', 'riccardo', 'zanotti', '4444444444'),
('a5', 'cesare', 'cremonini', '5555555555');
```

```
INSERT INTO `categorie` (`id_categoria`, `descrizione`) VALUES ('c1', 'sfilata'),
('c2', 'concerto'), ('c3', 'fiera'), ('c4', 'beach party');
```

```
INSERT INTO `membri_community` (`id_utente`, `nome`, `cognome`, `email`) VALUES
('user1', 'domenico', 'auriemma', 'domaur03@gmail.com'), ('user2', 'alessio', 'contini',
'leoblacksoul@gmail.com'),
('user3', 'simone', 'zimbaridi', 'simbsimy28@gmail.com'), ('user4', 'elia', 'pascarella',
'mychemicalelia@gmail.com'),
('user5', 'ilaria', 'vinciguerra', 'ilavinc@gmail.com');
```

```
INSERT INTO `eventi` (`id_evento`, `titolo`, `data_evento`, `luogo_evento`, `id_categoria`, `id_utente`)
VALUES
('event1', 'napolicomicon', '2021-06-01', 'napoli', 'c3', 'user1'),
('event2', 'metgala', '2022-06-02', 'caserta', 'c1', 'user3'),
('event3', 'cremonini tour', '2021-06-15', 'avellino', 'c2', 'user4'),
('event4', 'firmacopiezerocalcare', '2021-06-22', 'napoli', 'c3', 'user3'),
('event5', 'pinguini tattici nucleari tour', '2022-06-22', 'benevento', 'c2', 'user2');
```

```
INSERT INTO `eventi_artisti` (`id_evento`, `id_artista`) VALUES ('event1', 'a1'),
('event1', 'a3'), ('event2', 'a2'), ('event3', 'a5'), ('event4', 'a1'), ('event5', 'a4');
```

```
INSERT INTO `commenti` (`id_utente`, `id_evento`, `commento`, `rating`) VALUES
('user1', 'event2', 'osceno', '1'), ('user3', 'event2', 'stupendo', '5'),
('user3', 'event3', 'così così', '2'), ('user4', 'event4', 'bellissimo', '5'),
('user4', 'event5', 'bello', '5');
```

QUERY 1:

```
SELECT * FROM `eventi` WHERE data_evento < CURRENT_DATE ORDER BY luogo_evento DESC;
```

QUERY 2:

```
SELECT membri_community.id_utente, nome, cognome, email FROM commenti RIGHT JOIN  
membri_community ON  
commenti.id_utente = membri_community.id_utente WHERE commenti.commento IS NULL;
```

QUERY 3:

```
SELECT titolo, AVG(rating), id_categoria FROM commenti INNER JOIN eventi ON  
commenti.id_evento = eventi.id_evento GROUP BY commenti.id_evento  
ORDER BY id_categoria ASC, titolo ASC;
```

QUERY 4:

```
SELECT id_utente, nome, cognome, max(num_commenti) FROM  
(SELECT commenti.id_utente, nome, cognome, count(commenti.id_utente) AS num_commenti  
FROM `commenti` INNER JOIN membri_community ON  
commenti.id_utente = membri_community.id_utente  
GROUP BY id_utente ORDER BY num_commenti DESC) astbl;
```