



Università degli Studi di Salerno



Dipartimento di Ingegneria dell'Informazione ed Elettrica e  
Matematica Applicata

Corso di Laurea in Ingegneria Informatica

## **Basi di Dati 2023/2024** **Canale I-Z**

Project Work  
**Traccia N. Z26 - Surveys**

Gruppo n. **01 – IZ**

WP	Cognome e Nome	Matricola	e-mail	Responsabile
1	Sanchez Munoz, Gabriel	ERASMSIN 03655	g.sanchezmunoz@unisa.studenti.it	X
2	Corzo Galdo, Jaime	ERASMSIN 03737	j.corzogaldo@studenti.unisa.it	

# Anno accademico 2023-2024

## Sommario

<b>1. Descrizione della realtà di interesse</b>	<b>3</b>
<b>2. Analisi delle specifiche</b>	<b>4</b>
2.1. Glossario dei termini	4
2.2. Strutturazione dei requisiti in frasi	4
2.2.1. Frasi di carattere generale	4
2.2.2. Frasi relative a <...>	4
2.2.3. Frasi relative a <...>	4
2.2.4. Frasi relative a tipi specifici di <...>	4
2.3. Identificazione delle operazioni principali	4
<b>3. Progettazione Concettuale</b>	<b>5</b>
3.1. Schema Concettuale	5
3.1.1. Note sullo schema E-R [opzionale]	6
3.2. Design Pattern	6
3.2.1. Pattern Relazione Ternaria [Sostituire con il nome del pattern usato]	6
3.2.2. Pattern XXX	6
3.3. Dizionario dei Dati	7
3.4. Regole Aziendali	8
<b>4. Progettazione Logica</b>	<b>9</b>
4.1. Ristrutturazione Schema Concettuale	9
4.1.1. Analisi delle Prestazioni	9
4.2. Analisi delle ridondanze	9
4.2.1. Analisi della ridondanza 1: Esami sostenuti	10
4.2.2. Analisi della ridondanza 2: ...	10
4.3. Eliminazione delle generalizzazioni	11
4.3.1. Generalizzazione <i>Entita1</i>	11
4.3.2. Generalizzazione <i>Persona</i>	11
4.4. Partizionamento/Accorpamento Entità e Associazioni	11
4.5. Scelta degli identificatori principali	11
4.6. Schema ristrutturato finale	12
4.7. Schema logico	13
4.8. Documentazione dello schema logico	13
<b>5. Normalizzazione</b>	<b>14</b>

<b>6. Script Creazione e Popolamento Database</b>	<b>15</b>
<b>7. Query SQL</b>	<b>16</b>
7.1. Query con operatore di aggregazione e join: Titolo query	16
7.2. Query nidificata complessa: Titolo query	16
7.3. Query insiemistica: Titolo query	16
7.4. Altre query	16
7.4.1. Titolo Query	16
7.4.2. Titolo Query	16
<b>8. Viste</b>	<b>17</b>
8.1. Vista <i>TitoloVista</i>	17
8.1.1. Query con Vista: Titolo query	17
<b>9. Trigger</b>	<b>18</b>
9.1. Trigger inizializzazione: <i>TitoloTrigger</i>	18
9.2. Trigger per vincoli aziendali	18
9.2.1. Trigger1: TitoloTrigger	18
9.2.2. Trigger2: TitoloTrigger	18

# 1. Descrizione della realtà di interesse

Titolo: **Surveys**

*Si vuole progettare una base di dati per un servizio che consente di creare sondaggi online. Un sondaggio è creato da un organizzatore. L'organizzatore può essere una persona fisica o un'organizzazione. L'organizzatore deve creare un account per poter accedere al servizio e fornire i suoi dati principali. Un sondaggio è descritto da un titolo, una descrizione, un obiettivo, un tema/argomento, una lingua, e la data di inizio del sondaggio e di fine (entro quando deve essere completato). L'organizzatore può scegliere l'argomento o tema da una lista di argomenti disponibili oppure inserirlo se non presente. Un sondaggio è costituito da più domande. Una domanda può essere: a scelta multipla; risposta aperta; voto o valore su una scala. La domanda inoltre può essere obbligatoria o facoltativa. Per le domande a scelta multipla devono essere indicate le possibili risposte; per quella in cui è possibile inserire un valore oppure un voto, deve essere indicata la scala. L'utente che risponde al questionario deve fornire i propri dati. È importante memorizzare anche la data e l'orario in cui l'utente compila il questionario, e la durata di compilazione. Ovviamente vanno memorizzate le risposte date dagli utenti. L'organizzatore inoltre può decidere se il questionario può essere compilato da chiunque, oppure invitare gli utenti che possono compilarlo. In questo caso viene creata una campagna per il sondaggio, in cui devono essere indicati dall'organizzatore gli utenti che devono completarlo, fornendo almeno nome, cognome e indirizzo email degli utenti che possono completare il questionario. In questo caso è di interesse conoscere quanti utenti rispondono al questionario e quanti non lo compilano. Uno stesso questionario può essere oggetto di più campagne (ad esempio, può essere ripetuto ogni anno, mantenendo le stesse domande, ma cambiando l'insieme degli utenti a cui è destinato). Quindi uno stesso utente può rispondere più volte a uno stesso questionario per campagne differenti. Al termine del questionario, viene creato un report contenente una sintesi delle risposte fornite dagli utenti.*

## 1.1. Analisi della realtà di interesse

L'obiettivo del progetto è di realizzare un database per un servizio di sondaggi online. Considerando il suo organizzatore (persona fisica o organizzazione) e gli utenti che fanno il sondaggio.

In particolare, in questo progetto l'attenzione sarà focalizzata su il formato del sondaggio, esistono varie tipologie di domande (a scelta multipla; risposta aperta; voto o valore su una scala, inoltre obbligatoria o facoltativa), chi ha accesso per fare il sondaggio e come fornire questo e infine memorizzare i dati e accedere a loro.

Ciò richiederà la memorizzazione degli utenti e informazione sulla realizzazione del sondaggio per alla fine realizzare un report con una sintesi delle risposte.

## 2. Analisi delle specifiche

<b>Workpackage</b>	<b>Task</b>	<b>Responsabile</b>
<b>WP0</b>	Analisi delle specifiche	Intero Gruppo

### 2.1. Glossario dei termini

	<b>Termine</b>	<b>Descrizione</b>	<b>Sinonimi</b>	<b>Collegamenti</b>
<b>1</b>	Sondaggio	E descritto da un titolo, una descrizione, un obiettivo, un tema/argomento, una lingua, la data di inizio e di fine del sondaggio, e le domande	questionario	Organizzatore, Utente, Domanda, Argomento, report, campagna
<b>2</b>	Organizzatore	Persona fisica o un'organizzazione che può creare un'account per creare un sondaggio.	-	Sondaggio, account
<b>3</b>	Utente	E candidato a rispondere alle domande del questionario fornendo i suoi dati.	-	Sondaggio, campagna
<b>4</b>	Domanda	Presenti nel questionario, può essere: a scelta multipla, risposta aperta, voto o calore su una scala, inoltre obbligatoria o facoltativa.	-	Sondaggio, Report
<b>5</b>	Report	Contiene una sintesi delle risposte fornite dagli utenti	-	Sondaggio, report
<b>6</b>	Account	Serve per dare accesso all'organizzatore al servizio con i suoi dati principali	-	Organizzatore
<b>7</b>	Campagna	Serve per invitare gli utenti che possono compilare un sondaggio fornendo almeno nome, cognome e indirizzo email di questi utenti	-	Sondaggio, Utente

Tabella 1. Glossario dei Termini

### 2.2. Strutturazione dei requisiti in frasi

#### 2.2.1. Frasi di carattere generale

*Si vuole progettare una base di dati per un servizio che consente di creare sondaggi online.*

#### 2.2.2. Frasi relative a <Sondaggio>

Un sondaggio è descritto da un titolo, una descrizione, un obiettivo, un tema/argomento, una lingua, e la data di inizio del sondaggio e di fine (entro quando deve essere completato).

Un sondaggio è costituito da più domande.

#### 2.2.3. Frasi relative a <Organizzatore>

Un sondaggio è creato da un organizzatore.

L'organizzatore può essere una persona fisica o un'organizzazione.

L'organizzatore deve creare un account per poter accedere al servizio e fornire i suoi dati principali.

L'organizzatore inoltre può decidere se il questionario può essere compilato da chiunque, oppure invitare gli utenti che possono compilarlo

#### 2.2.4. Frasi relative a tipi specifici di <Utente>

L'utente che risponde al questionario deve fornire i propri dati. È importante memorizzare anche la data e l'orario in cui l'utente compila il questionario, e la durata di compilazione.

#### 2.2.5. Frasi relative a tipi specifici di <Domanda>

Una domanda può essere: a scelta multipla; risposta aperta; voto o valore su una scala. La domanda inoltre può essere obbligatoria o facoltativa. Per le domande a scelta multipla devono essere indicate le possibili risposte; per quella in cui è possibile inserire un valore oppure un voto, deve essere indicata la scala.

Ovviamente vanno memorizzate le risposte date dagli utenti.

#### 2.2.6. Frasi relative a tipi specifici di <Report>

Al termine del questionario, viene creato un report contenente una sintesi delle risposte fornite dagli utenti.

#### 2.2.7. Frasi relative a tipi specifici di <Account>

L'organizzatore deve creare un account per poter accedere al servizio e fornire i suoi dati principali.

#### 2.2.8. Frasi relative a tipi specifici di <Campagna>

In una campagna devono essere indicati dall'organizzatore gli utenti che devono completarlo, fornendo almeno nome, cognome e indirizzo email degli utenti che possono completare il questionario. In questo caso è di interesse conoscere quanti utenti rispondono al questionario e quanti non lo compilano.

Uno stesso questionario può essere oggetto di più campagne (ad esempio, può essere ripetuto ogni anno, mantenendo le stesse domande, ma cambiando l'insieme degli utenti a cui è destinato). Quindi uno stesso utente può rispondere più volte a uno stesso questionario per campagne differenti.

#### 2.2.9. Frasi relative a tipi specifici di <Argomento>

*L'organizzatore può scegliere l'argomento o tema da una lista di argomenti disponibili oppure inserirlo se non presente.*

### 2.3. Identificazione delle operazioni principali

**inserireSondaggio:** Inserisci un nuovo sondaggio descritto da un titolo, una descrizione, un obiettivo, un tema/argomento, una lingua, e la data di inizio del sondaggio. (50 volte al giorno)

**inserireDomanda:**

Inserire una domanda in un sondaggio, selezionando il tipo di domanda. (1000 volte al giorno)

**inserireOrganizzatore:**

Inserisci un nuovo organizzatore con la sua account, indicando se è persona fisica o un'organizzazione, e i suoi dati principali (10 volte al giorno)

**inserireRisposte:** Inserisci i risposte degli utenti, con i suoi dati principali, l'orario e la durata. (1000 volte al giorno)

**report:** Fare un report di un sondaggio (50 volte al giorno)

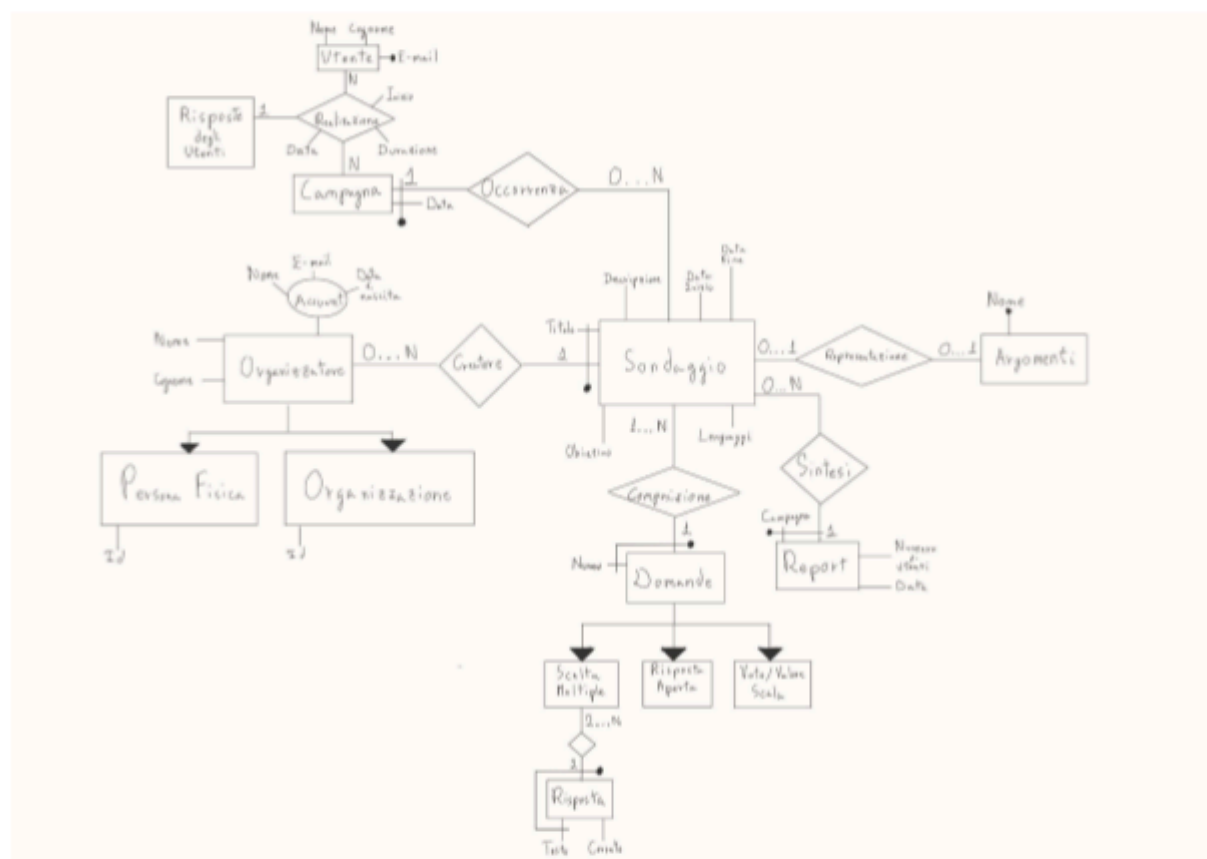
**inserireCampagna:** Inserirci una nuova campagna, con gli utenti che devono completarlo, fornendo almeno nome, cognome e indirizzo email (40 volte al giorno)

**inserireArgomento:** Inserirci un nuovo argomento. (1 volta al mese)

### 3. Progettazione Concettuale

Workpackage	Task	Responsabile
WP1	Progettazione Concettuale	Sanchez Munoz, Gabriel

#### 3.1. Schema Concettuale





## 3.2. Design Pattern

### 3.2.1. Pattern Relazione Ternaria

Abbiamo usato il pattern “Relazione Ternaria” per rappresentare l’interazione degli utenti con gli campagne, e le loro risposte. Queste sono tre entità relazionate tra loro.

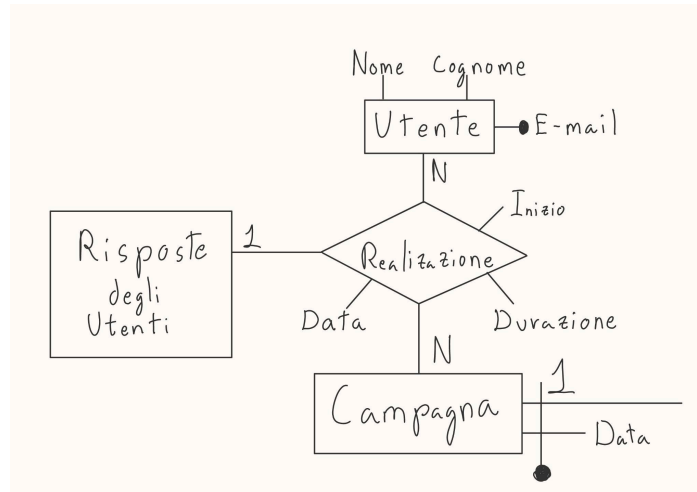


Figura 2. Schema del Pattern Relazione Ternaria

### 3.2.2. Pattern Relazione Ternaria

Abbiamo usato il pattern “parte di” con Domande e Sondaggio. Chiaramente Un sondaggio sta fatto di Domande, e le Domande sono parte di un sondaggio.

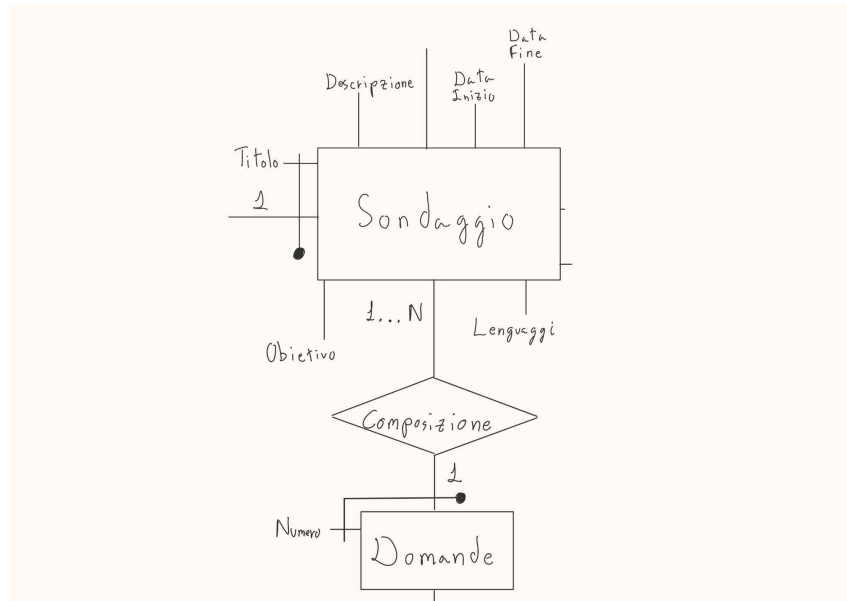
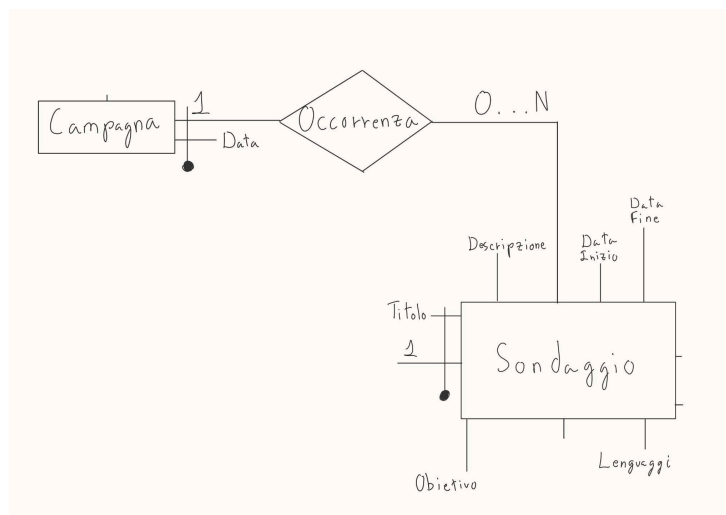


Figura 2. Schema del pattern Parte Di

### 3.2.3. Pattern instance of

In queste schema concettuale, sondaggio no e la realizzazione di un sondaggio, ma il suo schema. Campagna e il sondaggio “reale”, una istanza di sondaggio. Per questo motivo campagna ha il attributo Data e sondaggio no, e anche Campagna è relazionato con gli utenti che fanno il sondaggio.



### 3.3. Dizionario dei Dati

Entità	Descrizione	Attributi	Identificatore
Sondaggio	Sondaggio che crea l'organizzatore	titolo,una descrizione,un obiettivo,una lingua, e la data di inizio del sondaggio e di fine	Titolo + Organizzatore(identificatore esterno)
Organizzatore	L'organizzatore crea il sondaggio	Nome e Account(Atributo composto di nome di utente, email e data di nascita)	Account
Persona Fisica	Specializzazione di organizzatore	Proprietà di ereditarietà	Proprietà di ereditarietà
Organizzazione	Specializzazione di organizzatore	Proprietà di ereditarietà	Proprietà di ereditarietà
Utente	Chi deve fare il sondaggio	Nome, cognome ed email	email
Domanda	quesito del sondaggio	Testo della domanda, numero di domanda	Numero di domanda + sondaggio(identificatore esterno)
Scelta Multiple	Specializzazione di domanda	Proprietà di ereditarietà	Proprietà di ereditarietà
Risposta	Opzione di risposte della domanda a scelta multiple	Testo della risposta, corretta	Testo della risposta + scelta multiple(identificatore esterno)
Risposta Aperta	Specializzazione di domanda	Proprietà di ereditarietà	Proprietà di ereditarietà
Voto/Valore Scala	Specializzazione di domanda	Scala	Proprietà di ereditarietà
Campagna	Serve per invitare gli utenti che possono compilare un sondaggio	Data di inizio	Data di inizio + sondaggio(identificatore esterno)
Report	Creato alla fine del sondaggio con una sintesi delle risposte	Data di creazione	Data di creazione + sondaggio(identificatore esterno)
Argomento	Tema del sondaggio	Nome	Nome

Risposte di Utenti	Tutti i risposte degli utenti		Utente(identificatore esterno) + Campagna(identificatore esterno)
--------------------	-------------------------------	--	---

Tabella 2. Dizionario dei dati – Entità

Relazioni	Descrizione	Entità Coinvolte	Attributi
Creatore	Un organizzatore ha acceso a i sondaggi che a creato	Organizzatore Sondaggio	-
Sintesi	Un report presenta una sintesi delle risposte degli utenti	Sondaggio Report	-
Composizione	Le domande son parte di un sondaggio	Sondaggio Domande	-
Rappresentazione	Un tema e una descrizione breve di un sondaggio	Sondaggio Argomento	-
Realizzazione	Un utente risponde a un sondaggio (di una campagna)	Utente Risposte Campagna	Orario Durata Data
Occorrenza	Una campagna mette a disposizione degli utenti un sondaggio in una data	Campagna Sondaggio	-

Tabella 3. Dizionario dei dati - Relazioni

<b>Workpackage</b>	<b>Task</b>	<b>Responsabile</b>
<b>WP4</b>	Regole Aziendali	Corzo Galdo, Jaime

### 3.4. Regole Aziendali

<b>Regole di Vincolo</b>
<p><b>(RV1)</b> Un organizzatore NON PUÒ ESSERE alla stessa volta una persona fisica e una organizzazione.</p> <p><b>(RV2)</b> Uno stesso questionario PUÒ ESSERE oggetto di più campagne</p> <p><b>(RV3)</b> Uno stesso utente PUÒ RISPONDERE più volte a uno stesso questionario per campagne differenti.</p> <p><b>(RV4)</b> Se DEVONO MEMORIZZARE le risposte date dagli utenti.</p> <p><b>(RV5)</b> Alla fine del questionario se DEVE CREARE un report con una sintesi delle risposte fornite dagli utenti.</p> <p><b>(RV6)</b> Un organizzatore DEVE CREARE un'account per produrre sondaggi.</p> <p><b>(RV7)</b> Un organizzatore NON PUÒ AVERE più di un'account</p> <p><b>(RV8)</b> Il Nome di Account DEVE ESSERE unico</p> <p><b>(RV9)</b> Il Email di Account DEVE ESSERE unico</p> <p><b>(RV10)</b> I Temi che inserisce un'organizzatore che non ci sono sulla tabella di temi predifiniti DEVONO ESSERE INSERITI in questa tabella</p>

Tabella 4. Regole di vincolo

<b>Regole di derivazione</b>
<p><b>(RD1)</b> L'attributo N° utenti nella entità Report si ottiene contando il numero di occorrenze di Ruspostedegliutenti nell sondaggio a cui si riferisce il report.</p> <p><b>(RD2)</b> L'attributo durata nella entità Realizzazione si ottiene restando l'attributo orario inizio all'attributo orario fine</p>

Tabella 5. Regole di derivazione

## 4. Progettazione Logica

<b>Workpackage</b>	<b>Task</b>	<b>Responsabile</b>
<b>WP2</b>	Progettazione Logica	Corzo Galdo, Jaime

### 4.1. Ristrutturazione Schema Concettuale

#### 4.1.1. Analisi delle Prestazioni

##### 4.1.1.1. *Tavola dei volumi*

<b>Concetto</b>	<b>Tipo</b>	<b>Volume</b>
Sondaggio	E	500
Organizzatore	E	100
Persona Fisica	E	75
Organizzazione	E	25
Utente	E	45000 (250*100 + 50*4*100)
Domanda	E	25000
Scelta Multiple	E	10000
Risposta	E	30000
Risposta Aperta	E	5000
Voto/Valore Scala	E	10000
Campagna	E	2000
Report	E	2000
Argomento	E	10
Risposte di Utenti	E	250*1000 = 250.000
Creatore	R	500
Sintesi	R	2000
Composizione	R	25000
Rappresentazione	R	500
Realizzazione	R	250.000

Occorrenza	R	2000
------------	---	------

Tabella 6. Tavola dei volumi

#### 4.1.1.2. Tavola delle operazioni

Operazione	Tipo	Frequenza
inserireSondaggio	I	50 al giorno
inserireDomanda	I	1000 al giorno
inserireOrganizzatore	I	10 al giorno
inserireRisposte	I	1000 al giorno
report	B	50 al giorno
inserireCampagna	I	40 al giorno
inserireArgomento	I	1 al mese

Tabella 7. Tavola delle operazioni

## 4.2. Analisi delle ridondanze

- Sulla entità Report c'è un attributo derivabile di operazione di conteggio di occorrenza. N° utenti può essere derivato contando le occorrenze dell'associazione Realizzazione prendendo la campagna del sondaggio

Tipo: *Attributo Derivabile da conteggio di occorrenze*

### 4.2.1. Analisi della ridondanza 1: Numero Utenti

- Operazione 5: report

#### Tavola degli accessi in presenza di ridondanza

report			
Concetto	Costr.	Acc.	Tipo
Risposte degli Utenti	E	250	L
Sintesi	R	1	S
Report	E	1	S
Report	E	1	S

**Tavola degli accessi in assenza di ridondanza**

report			
Concetto	Costr.	Acc.	Tipo
Risposte degli Utenti	E	250	L
Sintesi	R	1	S
Report	E	1	S

#### **4.2.1.1. Valutazione della ridondanza 1**

Passiamo ora alla stima del costo delle operazioni. Prendiamo in considerazione soltanto la operazione 5 perché la ridondanza non è coinvolta nelle altre operazioni.

Vediamo che in presenza della ridondanza, la operazione 5 richiede 250 accessi(uno per ogni utente che in media compilano un sondaggio) in lettura all'entità Risposte degli utenti(per ottenere le risposte date dagli utenti nel questionario del report), come per la stima del costo le accessi alle entità/associazione nel percorso verso l'associazione Sintesi non cambiano non è necessario includerli. Poi, un accesso in scrittura all'associazione Sintesi(per memorizzare una nuova coppia Sondaggio-Report), e infine un accesso in scrittura all'entità Report(per creare il report) e un'altro accesso in scrittura all'entità Report(per aggiungere il n° di utenti).

Consideriamo ora il caso in cui il dato ridondante sia assente. L'operazione 5 richiede 250 accessi(uno per ogni utente) in lettura all'entità Risposte degli utenti(per ottenere le risposte date dagli utenti nel questionario del report). Poi, un accesso in scrittura all'associazione Sintesi(per memorizzare una nuova coppia Sondaggio-Report), e infine un accesso in scrittura all'entità Report(per creare il report).

Quindi abbiamo un attributo ridondante già che quando abbiamo questo attributo è necessario sempre un accesso di più per completare l'operazione 5.



### 4.3. Eliminazione delle generalizzazioni

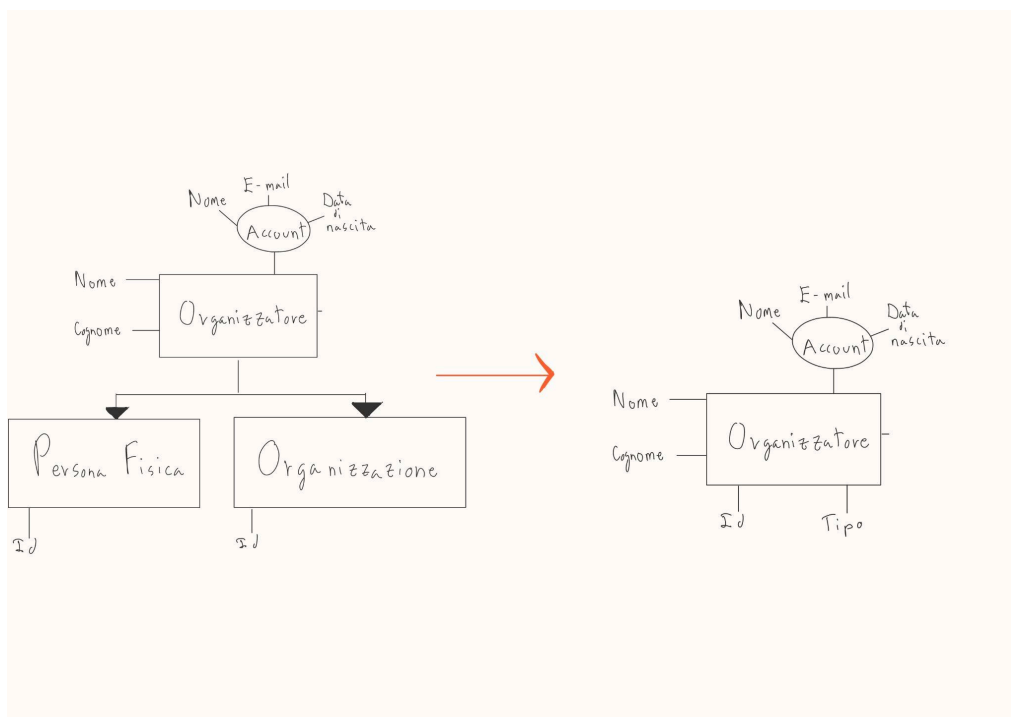
#### 4.3.1. Generalizzazione *Organizzatore*

La prima generalizzazione riguarda Organizzatore e le specializzazioni Persona Fisica e Organizzazione, per risolvere questa prendiamo l'accorpamento delle figlie della generalizzazione nel genitore. Prendiamo questa opzione perché tra Persona Fisica e Organizzazione non c'è molta differenza con le operazioni che devono fare e tra le sue attributi.

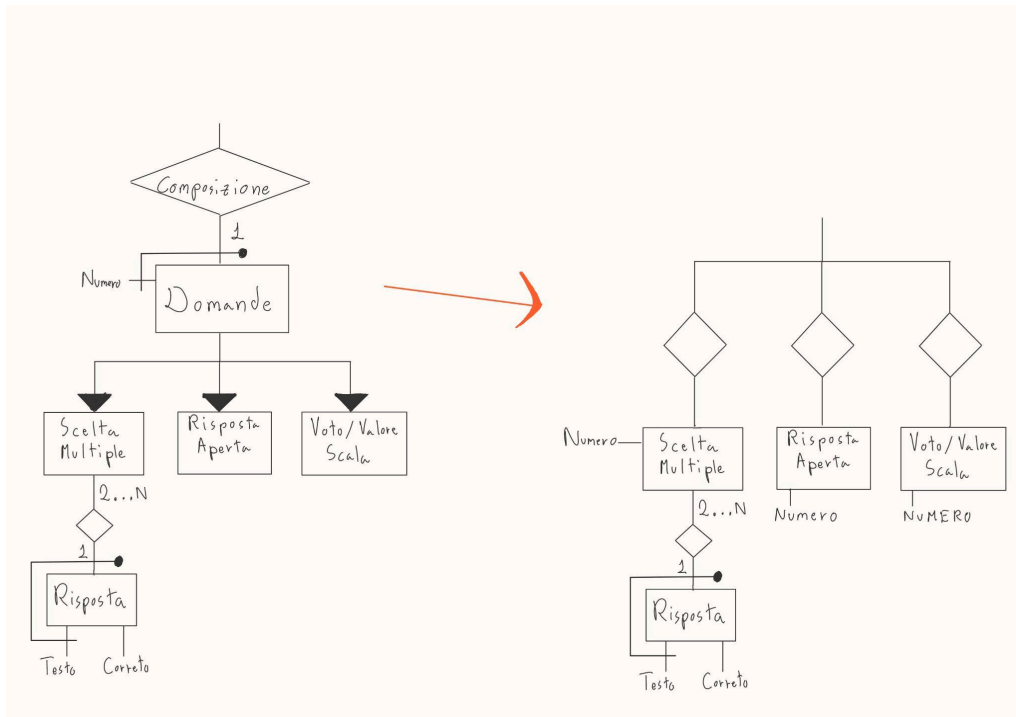
Quindi rimuoviamo Persona Fisica e Organizzazione e il suo ID aggiunto a Organizzazione. In Organizzazione aggiungiamo anche un attributo per distinguere tra Persona Fisica e Organizzazione, questo attributo è Tipo

#### 4.3.2. Generalizzazione *Domanda*

La seconda generalizzazione riguarda Domande e le specializzazioni Scelta Multiple, Risposta Aperta e Voto/Valore Scala. Per risolvere questa situazione prendiamo l'accorpamento del genitore della generalizzazione nelle figlie. Prendiamo questa opzione perché abbiamo una generalizzazione totale e perché abbiamo differenza tra le specializzazioni, per esempio Scelta Multiple ha un'associazione con Risposte.

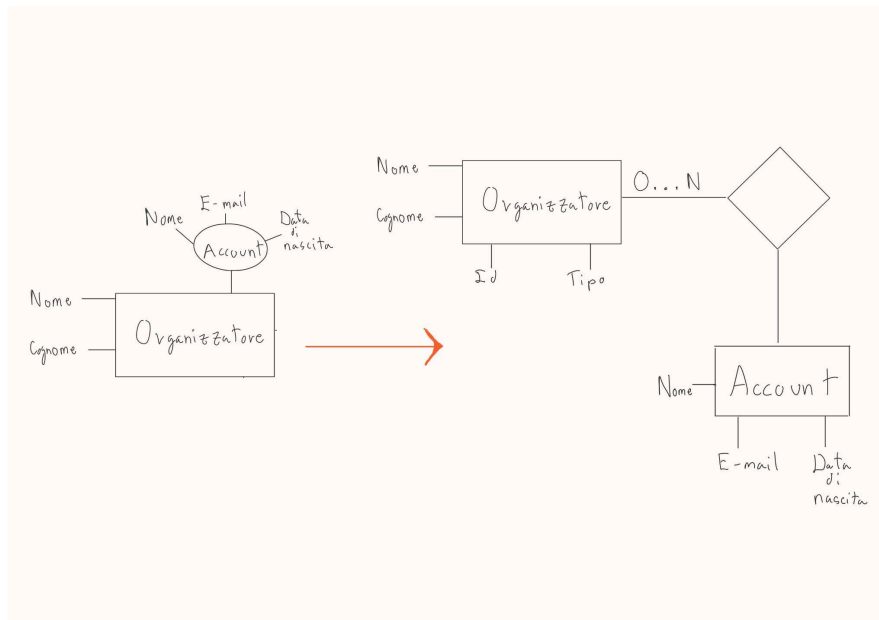


Quindi rimuoviamo Domande e per la proprietà dell'ereditarietà i suoi attributi, il suo identificatore, e la relazione a cui Domande partecipava vengono aggiunti a le sue figlie. In questo caso l'associazione Composizione e gli attributi Domanda, Obbligatorio e N° di domanda, che è l'identificatore esterno insieme alla entità Sondaggio.

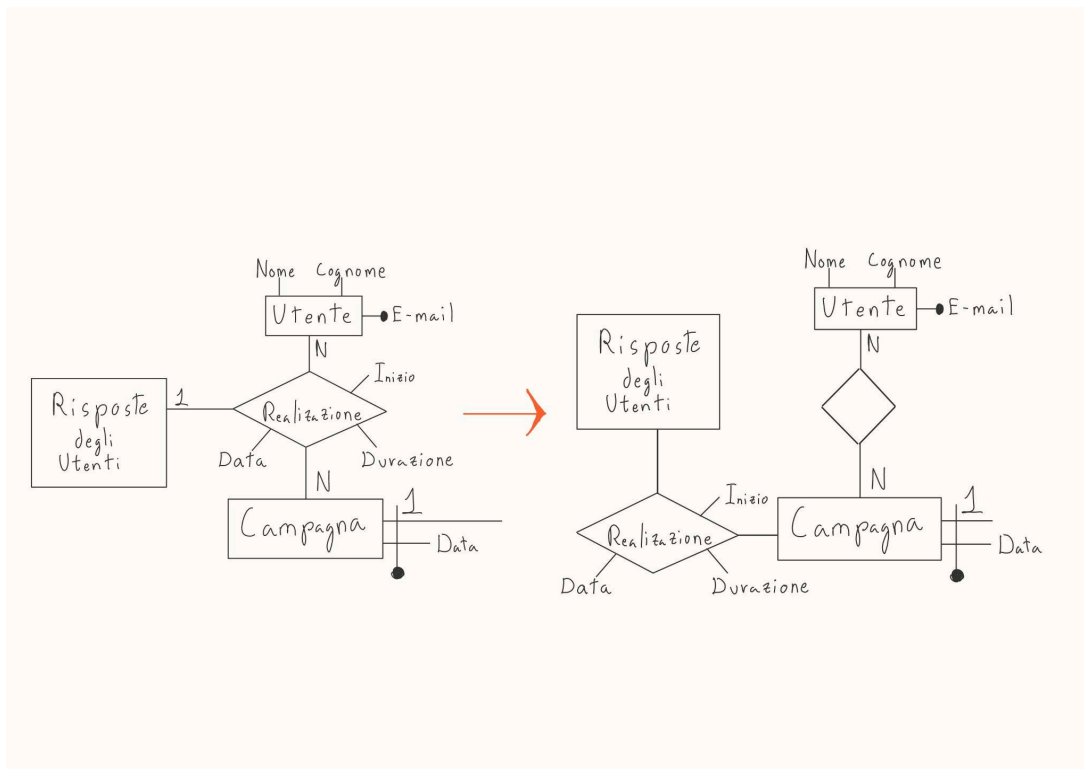


#### 4.4. Partizionamento/Accorpamento Entità e Associazioni

- Nel nostro schema E-R abbiamo l'entità Organizzatore che ha un attributo composto: Account, quindi andiamo a partizionare Organizzatore in la entità Organizzatore e la entità Account. Questo si chiama decomposizione verticale.



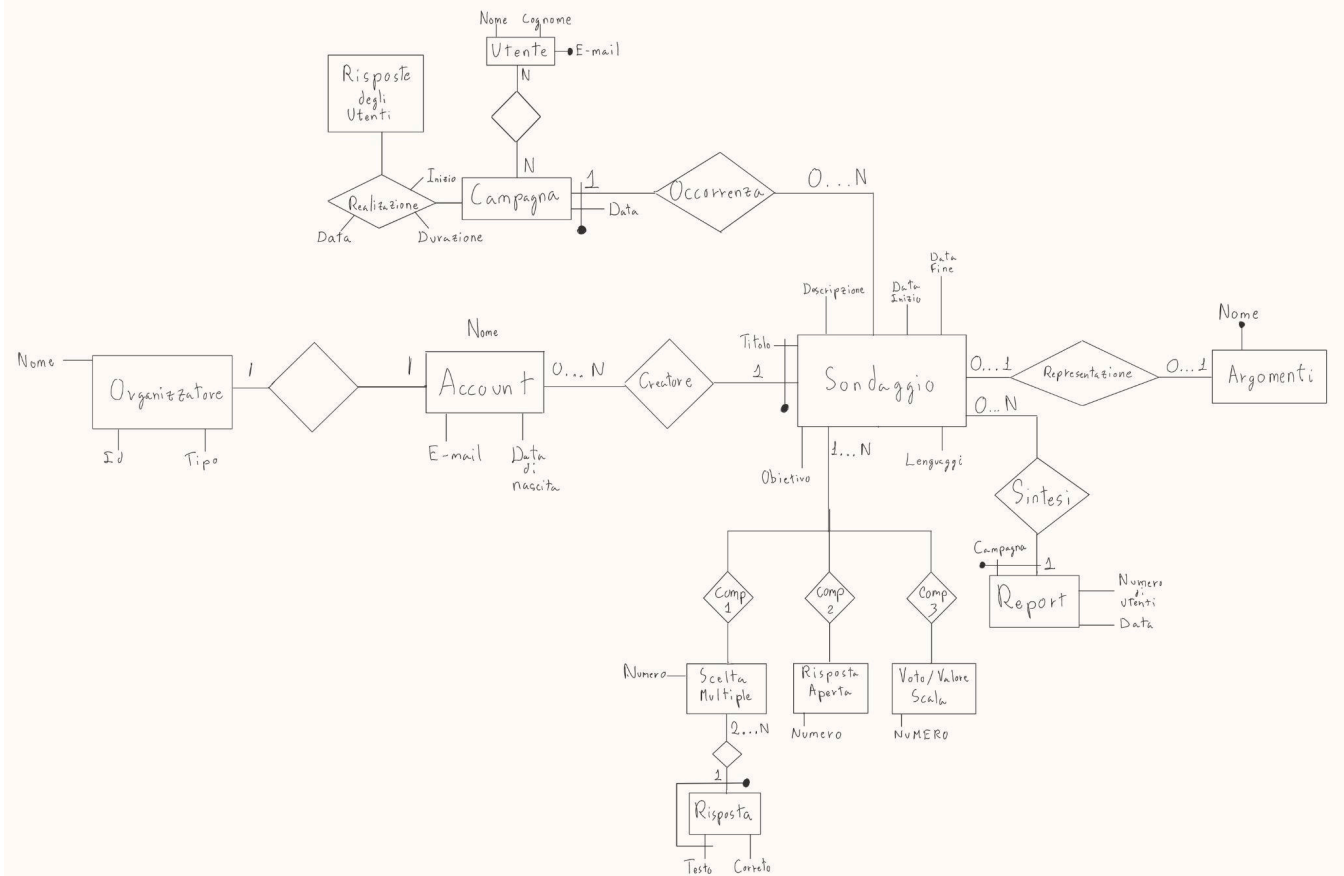
- Da un altro lato dobbiamo eliminare Risposte, che è un attributo multivalore di Scelta Multiple. Questo lo facciamo creando una nuova entità: Risposte, che ha due attributi: Testo e Corretta e aggiungiamo un'associazione tra Scelta Multiple e Risposte.



#### 4.5. Scelta degli identificatori principali

- Per l'identificatore di Risposte degli Utenti prendiamo l'attributo Email del utente e lo mettiamo come attributo e la entità Campagna.
- Per identificare Organizzazione abbiamo la opzione di tenere la entità Account oppure l'attributo ID. Ma non avere un identificatore esterno permette un risparmio di memoria e facilita l'operazione di Join. Quindi scegliamo l'attributo ID come identificatore.
- Per identificare Account prendiamo l'attributo ID che è lo stesso ID dell'Organizzatore che crea l'Account. Facciamo questo perché sebbene Email o Nome sono chiave primaria, già che non può avere due accounts con lo stesso Email o lo stesso nome, se facciamo la modifica di Email o il Nome dell'Account, dobbiamo modificare anche questo in tutte le entità che hanno Account come identificatore esterno.
- Per la proprietà della ereditarietà, Scelta Multiple, Risposta Aperta e Voto/Valore Scala hanno come identificatore l'attributo N° di domanda e la entità Sondaggio.

## 4.6. Schema ristrutturato finale



## 4.7. Schema logico

Utente(Email, Nome, Cognome)

Campagna(Data, Titolo Sondaggio, IDOrganizzatore)

Realizzazione(Utente, Campagna, Data, Orario inizio, Durata, Orario Fine) //En  
los trigger poner orarioinizio<orariofine)

Risposte Degli Utenti(EmailUtente, Data Campagna, Titolo Sondaggio,  
IDOrganizzatore, N° domanda, risposta)

Sondaggio(Titolo, IDOrganizzatore, Descrizione, Obiettivo,  
DataInizio, DataFine, Linguaggio, NomeTema)

Tema(Nome)

Report(Titolo Sondaggio, IDOrganizzatore, Data)

Scelte Multiple(Numero Domanda, Titolo Sondaggio, IDOrganizzatore,  
Domanda, Obbligatorio)

Risposta Aperta(Numero Domanda, Titolo Sondaggio, IDOrganizzatore,  
Domanda, Obbligatorio)

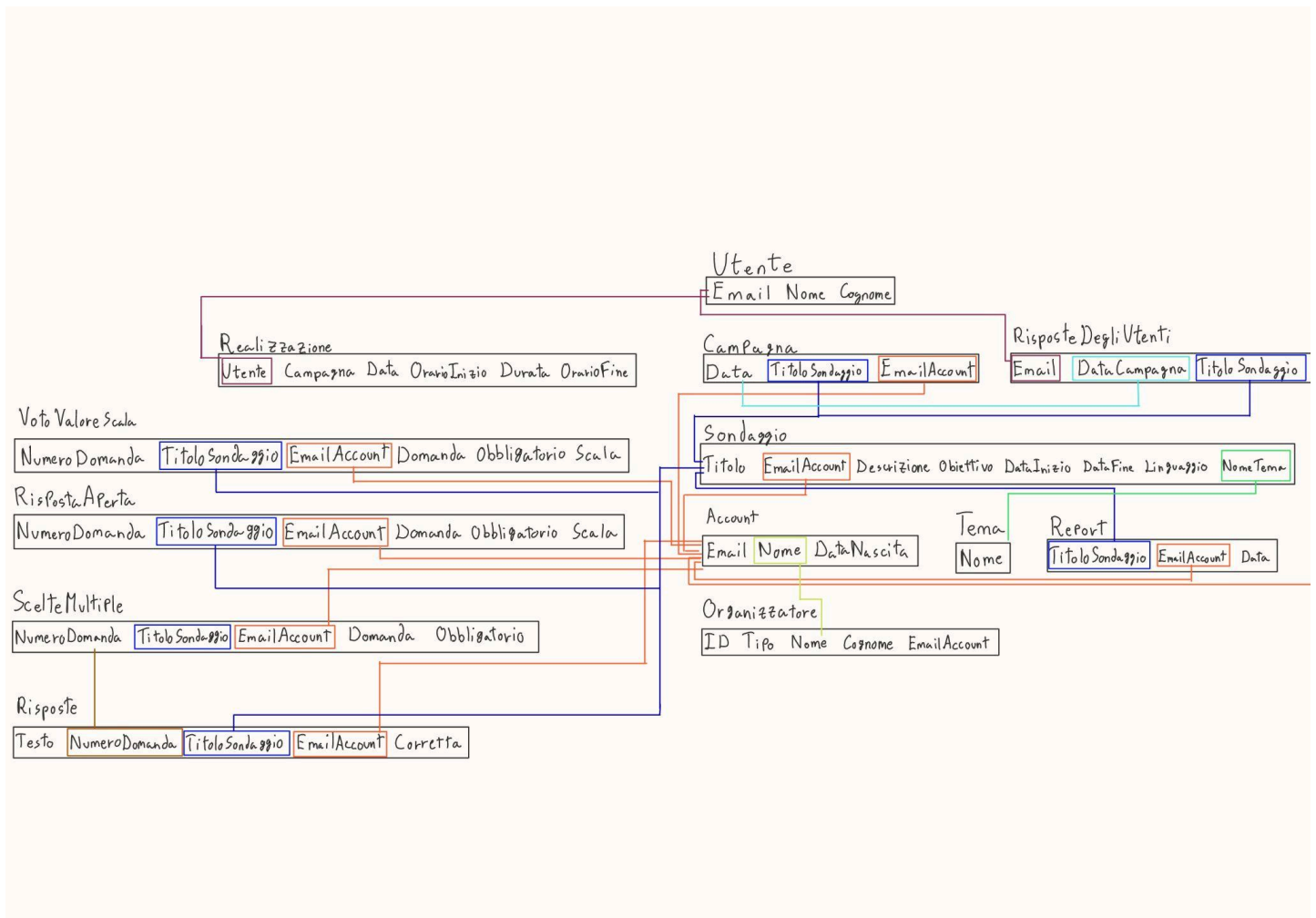
Voto/Valore Scala(Numero Domanda, Titolo Sondaggio, IDOrganizzatore,  
Domanda, Obbligatorio, Scala)

Risposte(Testo, Numero Domanda, Titolo Sondaggio, IDOrganizzatore,  
Corretta)

Account(IDOrganizzatore, Email, Nome, Data Nascita)

Organizzatore(ID, Tipo, Nome, EmailAccount)

## 4.8. Documentazione dello schema logico



## 5. Normalizzazione

<b>Workpackage</b>	<b>Task</b>	<b>Responsabile</b>
<b>WP3</b>	Normalizzazione	Sanchez Munoz, Gabriel

Anomalia di aggiornamento: Prima avevamo la entità Account con email come identificatore, quindi se noi aggiornavamo l'email e noi abbiamo un organizzatore con diverse sondaggio dovevamo aggiornare l'email in tutte le tuple. Ma ora abbiamo ID come chiave primaria di Account e quindi non abbiamo questo problema.

Account con email come identificatore

Organizzatore	Account	Sondaggio
Giuseppe	giuseppe@gmail.com	Sondaggio1
Giuseppe	giuseppe@gmail.com	Sondaggio2
Giuseppe	giuseppe@gmail.com	Sondaggio3
Vittorio	vittorio@gmail.com	Sondaggio4

Account con ID come identificatore

Organizzatore	Account	Sondaggio
Giuseppe	G01	Sondaggio1
Giuseppe	G01	Sondaggio2
Giuseppe	G01	Sondaggio3
Vittorio	V02	Sondaggio4

Anomalia di Cancellazione: Non ci sono

Anomalia di inserimento: Non ci sono

Terza Forma Normale

- Utente(**Email**, Nome, Cognome)

Utente è in terza forma normale, perché non ci sono dipendenze oltre que Email -> Nome e Email -> Cognome, e Email e una chiave primaria.

- Campagna(**Data, Titolo Sondaggio, IDAccount**)

Campagna è in terza forma normale perché non ci sono dipendenze.

- Realizzazione(**Utente, Campagna, Data, Orario inizio, Durata, Orario Fine**)

Realizzazione non è in terza forma normale, perché (Orario Inizio,Durata) -> Orario fine, ma né (Orario Inizio,Durata) è superchiave né Orario Fine è attributo primario.

Una possibile soluzione e creare i seguenti due tabelli:

R1(**Utente, Campagna, Data, Orario inizio, Durata**)

R2(**Orario inizio, Durata,Orario Fine**)

Ma crediamo che creare queste tabelle è eccessivo, quindi rimuoviamo l'orario finale e verrà calcolato con un'operazione.

- Risposte Degli Utenti(**Email, Data Campagna, Titolo Sondaggio, IDAccount**)

Risposte degli utenti è in terza forma normale perché non ci sono dipendenze.

- Sondaggio(**Titolo, IDAccount, Descrizione, Obiettivo, DataInizio, DataFine, Linguaggio, NomeTema** )

Sondaggio è in terza forma normale perché non ci sono dipendenza meno (Titolo, IDAccount) -> (resto di attributi).

- Tema(**Nome**)

Tema è in terza forma normale perché solo ha un atributo.

- Report(**Titolo Sondaggio, IDAccount, Data**)

Report è in terza forma normale perche solo è la dipendenza (Titolo Sondaggio, IDAccount) -> Data, dove (Titolo Sondaggio, IDAccount) e la chiave primaria.

- Scelte Multiple(**Numero Domanda, Titolo Sondaggio, IDAccount, Domanda, Obbligatorio**)

Scelte Multiple è in terza forma normale perché l'unica possibile dipendenza non banale sarebbe Domanda -> Obbligatorio ma possiamo avere un sondaggio in cui una Domanda sia Obbligatoria e un'altro Sondaggio in cui la stessa Domanda non sia obbligatoria quindi non è una dipendenza.



- Risposta Aperta(**Numero Domanda, Titolo Sondaggio, IDAccount, Domanda, Obbligatorio**)

Scelte Multiple è in terza forma normale perché l'unica possibile dipendenza non banale sarebbe Domanda -> Obbligatorio ma possiamo avere un sondaggio in cui una Domanda sia Obbligatoria e un'altro Sondaggio in cui la stessa Domanda non sia obbligatoria quindi non è una dipendenza.

- Voto/Valore Scala(**Numero Domanda, Titolo Sondaggio, IDAccount, Domanda, Obbligatorio, Scala**)

Scelte Multiple è in terza forma normale perché l'unica possibile dipendenza non banale sarebbe Domanda -> Obbligatorio ma possiamo avere un sondaggio in cui una Domanda sia Obbligatoria e un'altro Sondaggio in cui la stessa Domanda non sia obbligatoria quindi non è una dipendenza.

- Risposte(**Testo, Numero Domanda, Titolo Sondaggio, IDAccount, Corretta**)

Risposte è in terza forma normale perché l'unica possibile dipendenza non banale sarebbe Testo -> Corretta ma possiamo avere una domanda in cui il testo della risposta sia il corretto e un'altra domanda in cui lo stesso testo non sia il corretto quindi non è una dipendenza.

- Account(**ID, Email, Nome, Data Nascita**)

Account è in terza forma normale già che sebbene abbiamo la dipendenza Nome -> Data Nascita(per ogni Nome c'è una Data Nascita quindi sapendo il Nome sappiamo la Data Nascita), Nome è una chiave primaria, e quindi una superchiave. Il resto di dipendenze hanno Email, che è una chiave primaria, come elemento a sinistra.

- Organizzatore(**ID, Tipo, Nome, EmailAccount**)

Organizzatore è in terza forma normale già che non ci sono dipendenze dove l'elemento a sinistra non sia ID.

## Forma Normale di Boyce e Codd

Abbiamo arrivato alla terza forma normale, anzi abbiamo risolto tutte le dipendenze in modo che a sinistra di ogni dipendenza abbiamo una superchiave, quindi ci siamo nella Forma Normale di Boyce e Codd.

## 6. Script Creazione e Popolamento Database

<b>Workpackage</b>	<b>Task</b>	<b>Responsabile</b>
<b>WP2</b>	SQL: Script creazione e popolamento	Corzo Galdo, Jaime

Unset

```
set datestyle to European;
```

```
-- Database: test
```

```
-- DROP DATABASE IF EXISTS test;
```

```
CREATE DATABASE test
```

```
WITH
```

```
OWNER = postgres
```

```
ENCODING = 'UTF8'
```

```
LC_COLLATE = 'es_ES.UTF-8'
```

```
LC_CTYPE = 'es_ES.UTF-8'
```

```
LOCALE_PROVIDER = 'libc'
```

```
TABLESPACE = pg_default
```

```
CONNECTION LIMIT = -1
```

```
IS_TEMPLATE = False;
```

```
DROP VIEW IF EXISTS reportView;
```

```
DROP VIEW IF EXISTS numeroDomande;
```

```
DROP VIEW IF EXISTS numerodiutenti;
```

```
DROP VIEW IF EXISTS numerodicampagne;
```

```
DROP TABLE IF EXISTS scalaVotoDomande;
```

```
DROP TABLE IF EXISTS risposte;
```

```
DROP TABLE IF EXISTS domandeScelteMultiple;
```

```
DROP TABLE IF EXISTS report;
```

```
DROP TABLE IF EXISTS realizzazione;
```

```
DROP TABLE IF EXISTS domandeAperti;
```

```
DROP TABLE IF EXISTS risposteDegliUtenti;
```

```
DROP TABLE IF EXISTS campagna;
```

```
DROP TABLE IF EXISTS account;
```

```
DROP TABLE IF EXISTS sondaggio;
```

```
DROP TABLE IF EXISTS utente;
```

```
DROP TABLE IF EXISTS organizzatore;
```

```
DROP TABLE IF EXISTS tema;
```

```

CREATE TABLE IF NOT EXISTS utente
(
    nome            varchar(50) NOT NULL,
    cognome         varchar(50),
    email           varchar(50) PRIMARY KEY
);

insert into utente (Nome, Cognome, Email) values ('Garwin',
'Liissimore', 'glissimore0@army.mil');
insert into utente (Nome, Cognome, Email) values ('Sean', 'Shah',
'ssshah1@deviantart.com');
insert into utente (Nome, Cognome, Email) values ('Algernon',
'Edmondson', 'aedmondson2@nationalgeographic.com');
insert into utente (Nome, Cognome, Email) values ('Freeman',
'Teasdale', 'fteasdale3@wsj.com');
insert into utente (Nome, Cognome, Email) values ('Viki', 'Brankley',
'vbrankley4@go.com');
insert into utente (Nome, Cognome, Email) values ('Riccardo',
'Askell', 'raskell5@icio.us');
insert into utente (Nome, Cognome, Email) values ('Malchy',
'Carlesi', 'mcarlesi6@wiley.com');
insert into utente (Nome, Cognome, Email) values ('Ellen', 'Jankin',
'ejankin7@illinois.edu');
insert into utente (Nome, Cognome, Email) values ('Misha',
'MacBarron', 'mmacbarron8@theatlantic.com');
insert into utente (Nome, Cognome, Email) values ('Ciel', 'Heinsius',
'cheinsius9@skype.com');
insert into utente (Nome, Cognome, Email) values ('Alvy', 'Sprasen',
'asprasena@multiply.com');
insert into utente (Nome, Cognome, Email) values ('Ringo', 'Rainton',
'rraintonb@nps.gov');
insert into utente (Nome, Cognome, Email) values ('Binni', 'Rollin',
'brollinc@barnesandnoble.com');
insert into utente (Nome, Cognome, Email) values ('Carly',
'Jerratsch', 'cjerratschd@goodreads.com');
insert into utente (Nome, Cognome, Email) values ('Rhiamon',
'Sheppey', 'rsheppeye@twitter.com');
insert into utente (Nome, Cognome, Email) values ('Carey', 'Dukes',
'cdukesf@ezinearticles.com');
insert into utente (Nome, Cognome, Email) values ('Jamey', 'Bambury',
'jbamburyg@cloudflare.com');

```

```

        insert into utente (Nome, Cognome, Email) values ('Town', 'Dorre',
'tdorreh@time.com');
        insert into utente (Nome, Cognome, Email) values ('Thedric',
'Damrel', 'tdamreli@parallels.com');
        insert into utente (Nome, Cognome, Email) values ('Margi',
'Fieldgate', 'mfieldgatej@patch.com');
        insert into utente (Nome, Cognome, Email) values ('Rinaldo',
'Hilland', 'rhillandk@blinklist.com');
        insert into utente (Nome, Cognome, Email) values ('Fitzgerald',
'Cranny', 'fcrannyl@google.it');
        insert into utente (Nome, Cognome, Email) values ('Rosalinda',
'Cockshutt', 'rcockshuttm@google.com.br');
        insert into utente (Nome, Cognome, Email) values ('Sadie', 'Duley',
'sduleyn@lycos.com');
        insert into utente (Nome, Cognome, Email) values ('Karee', 'Sand',
'ksando@t-online.de');

```

```
CREATE TABLE IF NOT EXISTS organizzatore
```

```

(
    id          varchar(50) PRIMARY KEY,
    tipo        varchar(50) NOT NULL,
    nome        varchar(50) NOT NULL,
    email       varchar(26),

    UNIQUE (id, email),
    CONSTRAINT tipo_constraint CHECK(tipo = 'Organizzazione' OR tipo =
'Persona fisica')
);

```

```

        INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('a478fce5-8d11-45aa-a5e6-5cae6b44dc27','Organizzazione','Luettgen-Franeck
i','cfinlaison0@cnn.com');

```

```

        INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('34fce778-900e-4b7f-b756-d217f802ccde','Organizzazione','Zulauf
LLC','jlimer1@xinhuanet.com');

```

```

        INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('056acffa-a285-4890-9c25-4ce123fbbf35','Persona
fisica','Joscelin','jjahnke2@eventbrite.com');

```

```

        INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('0c3b1f6c-6878-4224-a91c-3952b2fc7775','Persona
fisica','Jewell','jburdess3@chronoengine.com');

```

```

INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('29926989-171a-484f-94bc-84eb2f549c4a','Persona
fisica','Claribel','cmidner4@digg.com');

INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('c16d45e5-92de-452a-bd3a-811c64649158','Organizzazione','Auer
LLC','cthirtle5@w3.org');

INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('7af28b7c-e7d8-48fc-b3ba-25fc88615c8e','Organizzazione','Mayer
Inc','lbernet6@mysql.com');

INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('0df773f4-b97e-4337-981b-d850268b1f1c','Organizzazione','Robel-Pagac','gd
ummett7@ovh.net');

INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('97d99bef-4e1d-4aa0-81e0-f81d341efdd6','Persona
fisica','Dory','dpartner8@imageshack.us');

INSERT INTO organizzatore(ID,Tipo,Nome,Email) VALUES
('a6b2a2ba-0029-45d3-a756-94c6b31a5ff7','Persona
fisica','Kerrill','klanigan9@yelp.com');

CREATE TABLE IF NOT EXISTS tema
(
    nome varchar(12) PRIMARY KEY
);

insert into tema (Nome) values ('pretium');
insert into tema (Nome) values ('neque');
insert into tema (Nome) values ('vulputate');
insert into tema (Nome) values ('eu');
insert into tema (Nome) values ('phasellus');
insert into tema (Nome) values ('vehicula');
insert into tema (Nome) values ('platea');
insert into tema (Nome) values ('augue');
insert into tema (Nome) values ('id');
insert into tema (Nome) values ('in');
insert into tema (Nome) values ('velit');
insert into tema (Nome) values ('fermentum');
insert into tema (Nome) values ('ligula');
insert into tema (Nome) values ('diam');

CREATE TABLE IF NOT EXISTS sondaggio
(
    titolo varchar(50) NOT NULL,
    id_organizzatore varchar(50) NOT NULL,

```

```

nome_tema          varchar(50),
descrizione        varchar(256),
obbiettivo         varchar(128),
data_inizio        date          NOT NULL,
data_fine          date,
linguaggio         varchar(16),

PRIMARY KEY (titolo, id_organizzatore),
FOREIGN KEY (id_organizzatore) REFERENCES organizzatore (id),
FOREIGN KEY (nome_tema) REFERENCES tema (nome)
);

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('A Man
Called Magnum', 'a478fce5-8d11-45aa-a5e6-5cae44dc27', 'pretium', 'nullam
sit amet', 'eu tincidunt in leo', '20/05/2020', '09/03/2023', 'Tamil');
insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values
('Carny', 'a478fce5-8d11-45aa-a5e6-5cae44dc27', 'pretium', 'in purus eu
magna vulputate', 'purus eu magna vulputate', '22/10/2022', '28/09/2023',
'Polish');
insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('Hail
Columbia!', 'a478fce5-8d11-45aa-a5e6-5cae44dc27', 'pretium', 'vestibulum
ante ipsum primis in', 'vel ipsum praesent blandit lacinia', '01/05/2022',
'22/06/2023', 'Romanian');
insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('War
and Peace (Voyna i mir)', '34fce778-900e-4b7f-b756-d217f802ccde', 'neque',
'sclerisque', 'cras mi pede malesuada in', '11/09/2021', '20/10/2023',
'Malay');
insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('All
About Eve', '34fce778-900e-4b7f-b756-d217f802ccde', 'neque', 'luctus rutrum
nulla tellus', 'at turpis donec posuere', '11/01/2023', '10/07/2023',
'Irish Gaelic');
insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values
('Cutting Edge: The Magic of Movie Editing, The',
'34fce778-900e-4b7f-b756-d217f802ccde', 'neque', 'sclerisque quam turpis
adipiscing', 'proin eu', '09/06/2022', '17/03/2023', 'Azeri');
insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('Venus

```

```

in Fur (La Vénus à la fourrure)', '056acffa-a285-4890-9c25-4ce123fbbf35',
'vulputate', 'luctus rutrum nulla', 'aliquam sit amet diam', '23/07/2022',
'03/12/2022', 'Khmer');

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values
('Stockholm Stories', '056acffa-a285-4890-9c25-4ce123fbbf35', 'pretium',
'ultrices libero', 'magna vulputate', '03/02/2022', '02/01/2023',
'Portuguese');

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('Down
to Earth', '056acffa-a285-4890-9c25-4ce123fbbf35', 'vulputate', 'maecenas
tincidunt lacus at', 'penatibus et magnis dis parturient', '30/12/2023',
'05/10/2024', 'Swati');

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('About
Alex', '29926989-171a-484f-94bc-84eb2f549c4a', 'vulputate', 'suscipit',
'vestibulum ante ipsum primis', '21/05/2022', '15/11/2023', 'Latvian');

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('Don''t
Tell Anyone (No se lo digas a nadie)',
'29926989-171a-484f-94bc-84eb2f549c4a', 'eu', 'massa', 'vulputate',
'01/01/2022', '19/02/2025', 'Dari');

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('The
Phantom Light', 'c16d45e5-92de-452a-bd3a-811c64649158', 'eu', 'sit', 'at
lorem integer tincidunt', '25/07/2020', '27/12/2023', 'English');

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values
('Suburbia', 'c16d45e5-92de-452a-bd3a-811c64649158', 'eu', 'vel enim sit',
'viverra eget', '15/05/2023', '12/07/2024', 'Macedonian');

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('Full
Moon in Blue Water', 'c16d45e5-92de-452a-bd3a-811c64649158', 'phasellus',
'odio donec vitae', 'mauris sit amet eros', '19/03/2021', '30/01/2024',
'Dhivehi');

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('Pool
Without Water, A (Mizu no nai puuru)',
'a6b2a2ba-0029-45d3-a756-94c6b31a5ff7', 'phasellus', 'velit eu est congue
elementum', 'penatibus et', '15/08/2022', '18/02/2023', 'Sotho');

insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values
('Tucker: The Man and His Dream', 'a6b2a2ba-0029-45d3-a756-94c6b31a5ff7',

```

```
'phasellus', 'vestibulum', 'proin leo odio porttitor', '25/10/2022',
'18/10/2023', 'Tswana');
insert into sondaggio (Titolo, ID_organizzatore, Nome_tema,
Descrizione, Obbiettivo, Data_inizio, Data_fine, Linguaggio) values ('Female
Agents (Les femmes de l'ombre)', 'a6b2a2ba-0029-45d3-a756-94c6b31a5ff7',
'phasellus', 'fermentum justo', 'non sodales', '08/12/2021', '04/02/2024',
'Arabic');
```

```
CREATE TABLE IF NOT EXISTS campagna
```

```
(
    data            date            NOT NULL,
    titolo_sondaggio varchar(50) NOT NULL,
    id_account      varchar(50) NOT NULL,

    PRIMARY KEY (data, titolo_sondaggio, id_account),
    FOREIGN KEY (titolo_sondaggio, id_account) REFERENCES sondaggio
(titolo, id_organizzatore)
);
```

```
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('22/01/2021', 'A Man Called Magnum',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27');
```

```
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('24/11/2021', 'A Man Called Magnum',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27');
```

```
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('12/03/2021', 'A Man Called Magnum',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27');
```

```
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('01/05/2021', 'A Man Called Magnum',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27');
```

```
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('01/08/2023', 'Carny', 'a478fce5-8d11-45aa-a5e6-5caebb44dc27');
```

```
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('01/05/2023', 'Carny', 'a478fce5-8d11-45aa-a5e6-5caebb44dc27');
```

```
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('24/06/2022', 'Hail Columbia!', 'a478fce5-8d11-45aa-a5e6-5caebb44dc27');
```

```
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('24/02/2022', 'War and Peace (Voyna i mir)',
'34fce778-900e-4b7f-b756-d217f802ccde');
```



```

insert into campagna (Data, Titolo_sondaggio, ID_account) values
('13/06/2023','All About Eve', '34fce778-900e-4b7f-b756-d217f802ccde');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('19/07/2022','Cutting Edge: The Magic of Movie Editing, The',
'34fce778-900e-4b7f-b756-d217f802ccde');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('13/08/2022','Cutting Edge: The Magic of Movie Editing, The',
'34fce778-900e-4b7f-b756-d217f802ccde');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('09/12/2022','Cutting Edge: The Magic of Movie Editing, The',
'34fce778-900e-4b7f-b756-d217f802ccde');

insert into campagna (Data, Titolo_sondaggio, ID_account) values
('03/08/2022','Venus in Fur (La Vénus à la fourrure)',
'056acffa-a285-4890-9c25-4ce123fbbf35');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('30/03/2022','Stockholm Stories',
'056acffa-a285-4890-9c25-4ce123fbbf35');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('05/01/2024','Down to Earth', '056acffa-a285-4890-9c25-4ce123fbbf35');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('27/06/2022','About Alex', '29926989-171a-484f-94bc-84eb2f549c4a');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('29/09/2022','Pool Without Water, A (Mizu no nai puuru)',
'a6b2a2ba-0029-45d3-a756-94c6b31a5ff7');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('22/12/2022','Pool Without Water, A (Mizu no nai puuru)',
'a6b2a2ba-0029-45d3-a756-94c6b31a5ff7');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('18/11/2022','Pool Without Water, A (Mizu no nai puuru)',
'a6b2a2ba-0029-45d3-a756-94c6b31a5ff7');

insert into campagna (Data, Titolo_sondaggio, ID_account) values
('13/09/2023','Tucker: The Man and His Dream',
'a6b2a2ba-0029-45d3-a756-94c6b31a5ff7');
insert into campagna (Data, Titolo_sondaggio, ID_account) values
('11/04/2022','Female Agents (Les femmes de l''ombre)',
'a6b2a2ba-0029-45d3-a756-94c6b31a5ff7');

```

```

CREATE TABLE IF NOT EXISTS scalaVotoDomande
(
    numerodomanda    integer    NOT NULL,
    titolosondaggio   varchar(50) NOT NULL,

```

```

    idorganizzatore    varchar(50) NOT NULL,
    domanda            varchar(100)    NOT NULL,
    obbligatorio        boolean    NOT NULL,
    scala              varchar(10) NOT NULL,

    PRIMARY KEY (idorganizzatore, titolosondaggio, numerodomanda),
    FOREIGN KEY (titolosondaggio, idorganizzatore) REFERENCES sondaggio
(titolo, id_organizzatore)
);

```

```

insert into ScalaVotoDomande (NumeroDomanda, TitoloSondaggio,
idorganizzatore, Domanda, Obbligatorio, Scala) values (1, 'Carny',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 'consectetuer eget rutrum at
lorem?', false, 'NotNumeric');
insert into ScalaVotoDomande (NumeroDomanda, TitoloSondaggio,
idorganizzatore, Domanda, Obbligatorio, Scala) values (2, 'Carny',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 'posuere cubilia curae duis
faucibus accumsan odio curabitur convallis duis consequat dui nec nisi?',
true, 'NotNumeric');
insert into ScalaVotoDomande (NumeroDomanda, TitoloSondaggio,
idorganizzatore, Domanda, Obbligatorio, Scala) values (1, 'Hail Columbia!',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 'porta volutpat erat quisque erat
eros viverra eget congue eget semper?', false, 'Numeric');
insert into ScalaVotoDomande (NumeroDomanda, TitoloSondaggio,
idorganizzatore, Domanda, Obbligatorio, Scala) values (2, 'Hail Columbia!',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 'nulla neque libero convallis eget
eleifend?', false, 'Numeric');
insert into ScalaVotoDomande (NumeroDomanda, TitoloSondaggio,
idorganizzatore, Domanda, Obbligatorio, Scala) values (1, 'All About Eve',
'34fce778-900e-4b7f-b756-d217f802ccde', 'non sodales sed tincidunt eu
felis?', true, 'NotNumeric');
insert into ScalaVotoDomande (NumeroDomanda, TitoloSondaggio,
idorganizzatore, Domanda, Obbligatorio, Scala) values (2, 'All About Eve',
'34fce778-900e-4b7f-b756-d217f802ccde', 'at dolor quis odio consequat
varius integer ac leo pellentesque ultrices mattis odio?', false,
'NotNumeric');
insert into ScalaVotoDomande (NumeroDomanda, TitoloSondaggio,
idorganizzatore, Domanda, Obbligatorio, Scala) values (1, 'Cutting Edge:
The Magic of Movie Editing, The', '34fce778-900e-4b7f-b756-d217f802ccde',
'et magnis dis parturient montes nascetur ridiculus mus etiam vel augue
vestibulum rutrum?', true, 'Numeric');
insert into ScalaVotoDomande (NumeroDomanda, TitoloSondaggio,
idorganizzatore, Domanda, Obbligatorio, Scala) values (2, 'Cutting Edge:

```

```
The Magic of Movie Editing, The', '34fce778-900e-4b7f-b756-d217f802ccde',
'et magnis dis parturient montes nascetur ridiculus mus etiam vel augue
vestibulum rutrum?', true, 'Numeric');
```

```
CREATE TABLE IF NOT EXISTS domandeScelteMultiple
(
    numerodomanda      integer      NOT NULL,
    titolosondaggio    varchar(50) NOT NULL,
    idorganizzatore    varchar(50) NOT NULL,
    domanda            text         NOT NULL,
    obbligatorio        boolean     NOT NULL,

    PRIMARY KEY (numerodomanda, titolosondaggio, idorganizzatore),
    FOREIGN KEY (titolosondaggio, idorganizzatore) REFERENCES sondaggio
(titolo, id_organizzatore)
);
```

```
insert into DomandeScelteMultiple (NumeroDomanda, TitoloSondaggio,
idOrganizzatore, Domanda, Obbligatorio) values (1, 'The Phantom
Light','c16d45e5-92de-452a-bd3a-811c64649158', 'risus semper porta volutpat
quam?', false);
```

```
insert into DomandeScelteMultiple (NumeroDomanda, TitoloSondaggio,
idOrganizzatore, Domanda, Obbligatorio) values (1,
'Suburbia','c16d45e5-92de-452a-bd3a-811c64649158', 'enim leo rhoncus sed
vestibulum sit?', true);
```

```
insert into DomandeScelteMultiple (NumeroDomanda, TitoloSondaggio,
idOrganizzatore, Domanda, Obbligatorio) values (1, 'Full Moon in Blue
Water','c16d45e5-92de-452a-bd3a-811c64649158', 'erat nulla tempus vivamus
in felis eu sapien cursus vestibulum proin eu mi nulla?', false);
```

```
insert into DomandeScelteMultiple (NumeroDomanda, TitoloSondaggio,
idOrganizzatore, Domanda, Obbligatorio) values (1, 'Stockholm
Stories','056acffa-a285-4890-9c25-4ce123fbbf35', 'ac leo pellentesque
ultrices mattis?', true);
```

```
insert into DomandeScelteMultiple (NumeroDomanda, TitoloSondaggio,
idOrganizzatore, Domanda, Obbligatorio) values (2, 'Stockholm
Stories','056acffa-a285-4890-9c25-4ce123fbbf35', 'cubilia curae nulla
dapibus dolor vel est donec odio justo sollicitudin ut?', false);
```

```
insert into DomandeScelteMultiple (NumeroDomanda, TitoloSondaggio,
idOrganizzatore, Domanda, Obbligatorio) values (1, 'About
Alex','29926989-171a-484f-94bc-84eb2f549c4a', 'placerat praesent blandit
nam nulla integer?', true);
```

```
insert into DomandeScelteMultiple (NumeroDomanda, TitoloSondaggio,
idOrganizzatore, Domanda, Obbligatorio) values (1, 'A Man Called
```

```
Magnum','a478fce5-8d11-45aa-a5e6-5caebb44dc27', 'a suscipit nulla elit ac
nulla sed?', true);
```

```
CREATE TABLE IF NOT EXISTS risposte
(
    testo                varchar(128)      NOT NULL,
    numerodomanda        integer          NOT NULL,
    titolosondaggio       varchar(50) NOT NULL,
    idorganizzatore       varchar(50) NOT NULL,
    corretta              boolean          NOT NULL,

    PRIMARY KEY (testo),
    FOREIGN KEY (numerodomanda, titolosondaggio, idorganizzatore)
REFERENCES domandeScelteMultiple (numerodomanda, titolosondaggio,
idorganizzatore)
);
```

```
CREATE TABLE IF NOT EXISTS report
(
    titolo_sondaggio     varchar(50) NOT NULL,
    iD_Organizzatore     varchar(50) NOT NULL,
    data                 date,

    PRIMARY KEY (titolo_sondaggio, iD_Organizzatore),
    FOREIGN KEY (titolo_sondaggio, iD_Organizzatore) REFERENCES
sondaggio (titolo, iD_Organizzatore)
);
```

```
insert into report (Titolo_sondaggio, ID_organizzatore, Data) values
('A Man Called Magnum', 'a478fce5-8d11-45aa-a5e6-5caebb44dc27',
'09/03/2023');
```

```
insert into report (Titolo_sondaggio, ID_organizzatore, Data) values
('Carny', 'a478fce5-8d11-45aa-a5e6-5caebb44dc27', '28/09/2023');
```

```
insert into report (Titolo_sondaggio, ID_organizzatore, Data) values
('Hail Columbia!', 'a478fce5-8d11-45aa-a5e6-5caebb44dc27', '22/06/2023');
```

```
insert into report (Titolo_sondaggio, ID_organizzatore, Data) values
('War and Peace (Voyna i mir)', '34fce778-900e-4b7f-b756-d217f802ccde',
'20/10/2023');
```

```
insert into report (Titolo_sondaggio, ID_organizzatore, Data) values
('All About Eve', '34fce778-900e-4b7f-b756-d217f802ccde', '10/07/2023');
```

```
insert into report (Titolo_sondaggio, ID_organizzatore, Data) values
('Cutting Edge: The Magic of Movie Editing, The',
'34fce778-900e-4b7f-b756-d217f802ccde', '17/03/2023');
```

```
insert into report (Titolo_sondaggio, ID_organizzatore, Data) values
('Venus in Fur (La Vénus à la fourrure)',
'056acffa-a285-4890-9c25-4ce123fbbf35', '3/12/2022');
```

```
CREATE TABLE IF NOT EXISTS realizzazione
(
    email                varchar(50) NOT NULL,
    data_campagna         date         NOT NULL,
    titolo_sondaggio      varchar(50) NOT NULL,
    id_organizzatore      varchar(50) NOT NULL,
    data                  date         NOT NULL,
    orario_inizio         time         NOT NULL,
    orario_fine           time         NOT NULL,
    durata                time,

    PRIMARY KEY (email, data_campagna, titolo_sondaggio,
id_organizzatore),
    FOREIGN KEY (data_campagna, titolo_sondaggio, id_organizzatore)
REFERENCES campagna (data, titolo_sondaggio, id_account),
    FOREIGN KEY (email) REFERENCES utente (email),

    CONSTRAINT datalogic CHECK (data >= data_campagna)
);

insert into realizzazione (Email, Data_campagna, Titolo_sondaggio,
ID_organizzatore, Data, orario_inizio, orario_fine, Durata) values
('glissimore0@army.mil', '22/01/2021', 'A Man Called Magnum',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', '12/11/2022', '10:12', '12:12',
'2:00');

insert into realizzazione (Email, Data_campagna, Titolo_sondaggio,
ID_organizzatore, Data, orario_inizio, orario_fine, Durata) values
('aedmondson2@nationalgeographic.com', '22/01/2021', 'A Man Called Magnum',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', '12/11/2022', '10:12', '12:12',
'2:00');

insert into realizzazione (Email, Data_campagna, Titolo_sondaggio,
ID_organizzatore, Data, orario_inizio, orario_fine, Durata) values
('sshah1@deviantart.com', '22/01/2021', 'A Man Called Magnum',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', '12/11/2022', '10:12', '12:12',
'2:00');
```

```
insert into realizzazione (Email, Data_campagna, Titolo_sondaggio,
ID_organizzatore, Data, orario_inizio, orario_fine, Durata) values
('glissimore0@army.mil', '01/08/2023', 'Carny',
'a478fce5-8d11-45aa-a5e6-5cae5b44dc27', '30/09/2023', '14:00', '18:26',
'2:26');
```

```
insert into realizzazione (Email, Data_campagna, Titolo_sondaggio,
ID_organizzatore, Data, orario_inizio, orario_fine, Durata) values
('sshah1@deviantart.com', '24/06/2022', 'Hail Columbia!',
'a478fce5-8d11-45aa-a5e6-5cae5b44dc27', '03/07/2022', '23:45', '23:57',
'0:12');
```

```
insert into realizzazione (Email, Data_campagna, Titolo_sondaggio,
ID_organizzatore, Data, orario_inizio, orario_fine, Durata) values
('sshah1@deviantart.com', '24/02/2022', 'War and Peace (Voyna i mir)',
'34fce778-900e-4b7f-b756-d217f802ccde', '06/06/2023', '3:25', '4:00',
'0:35');
```

```
insert into realizzazione (Email, Data_campagna, Titolo_sondaggio,
ID_organizzatore, Data, orario_inizio, orario_fine, Durata) values
('aedmondson2@nationalgeographic.com', '13/06/2023', 'All About Eve',
'34fce778-900e-4b7f-b756-d217f802ccde', '09/07/2023', '1:50', '16:50',
'15:00');
```

```
insert into realizzazione (Email, Data_campagna, Titolo_sondaggio,
ID_organizzatore, Data, orario_inizio, orario_fine, Durata) values
('aedmondson2@nationalgeographic.com', '19/07/2022', 'Cutting Edge: The
Magic of Movie Editing, The', '34fce778-900e-4b7f-b756-d217f802ccde',
'20/02/2023', '9:00', '20:40', '11:40');
```

```
CREATE TABLE IF NOT EXISTS domandeAperti
(
    numerodomanda      integer      NOT NULL,
    titolosondaggio     varchar(50) NOT NULL,
    idorganizzatore     varchar(50) NOT NULL,
    domanda             text        NOT NULL,
    obbligatorio        boolean     NOT NULL,

    PRIMARY KEY (idorganizzatore, titolosondaggio, numerodomanda),
    FOREIGN KEY (titolosondaggio, idorganizzatore) REFERENCES sondaggio
(titolo, id_organizzatore)
);
```

```
insert into DomandeAperti (NumeroDomanda, TitoloSondaggio, idOrganizzatore,
Domanda, Obbligatorio) values (1, 'War and Peace (Voyna i mir)',
```

```

'34fce778-900e-4b7f-b756-d217f802ccde', 'luctus rutrum nulla tellus in
sagittis dui vel nisl dui ac?', true);
insert into DomandeAperti (NumeroDomanda, TitoloSondaggio, idOrganizzatore,
Domanda, Obbligatorio) values (2, 'War and Peace (Voyna i mir)',
'34fce778-900e-4b7f-b756-d217f802ccde', 'nunc rhoncus dui vel sem sed
sagittis nam congue risus?', false);
insert into DomandeAperti (NumeroDomanda, TitoloSondaggio, idOrganizzatore,
Domanda, Obbligatorio) values (1, 'A Man Called Magnum',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 'sed tincidunt eu felis fusce
posuere felis sed?', false);
insert into DomandeAperti (NumeroDomanda, TitoloSondaggio, idOrganizzatore,
Domanda, Obbligatorio) values (1, 'Venus in Fur (La Vénus à la fourrure)',
'056acffa-a285-4890-9c25-4ce123fbbf35', 'nunc viverra dapibus nulla
suscipit ligula in lacus?', false);
insert into DomandeAperti (NumeroDomanda, TitoloSondaggio, idOrganizzatore,
Domanda, Obbligatorio) values (2, 'Venus in Fur (La Vénus à la fourrure)',
'056acffa-a285-4890-9c25-4ce123fbbf35', 'phasellus sit amet erat nulla?',
true);
insert into DomandeAperti (NumeroDomanda, TitoloSondaggio, idOrganizzatore,
Domanda, Obbligatorio) values (3, 'Venus in Fur (La Vénus à la fourrure)',
'056acffa-a285-4890-9c25-4ce123fbbf35', 'diam neque vestibulum eget
vulputate ut ultrices vel augue vestibulum ante?', true);
insert into DomandeAperti (NumeroDomanda, TitoloSondaggio, idOrganizzatore,
Domanda, Obbligatorio) values (1, 'Down to Earth',
'056acffa-a285-4890-9c25-4ce123fbbf35', 'et eros vestibulum ac est?',
false);
insert into DomandeAperti (NumeroDomanda, TitoloSondaggio, idOrganizzatore,
Domanda, Obbligatorio) values (2, 'Down to Earth',
'056acffa-a285-4890-9c25-4ce123fbbf35', 'libero nam dui proin leo odio
porttitor id consequat?', true);

```

```

CREATE TABLE IF NOT EXISTS risposteDegliUtenti

```

```

(
    email_utente    varchar(50) NOT NULL,
    data_campagna   date          NOT NULL,
    titolo_sondaggio varchar(50) NOT NULL,
    id_organizzatore varchar(50) NOT NULL,
    numeroDomanda   integer        NOT NULL,
    risposta        varchar(256) NOT NULL,

    PRIMARY KEY (email_utente, data_campagna, titolo_sondaggio,
id_organizzatore, numeroDomanda),

```

```

    FOREIGN KEY (data_campagna, titolo_sondaggio, id_organizzatore)
REFERENCES campagna (data, titolo_sondaggio, id_account),
    FOREIGN KEY (email_utente) REFERENCES utente (email)
);

insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('glissimore0@army.mil', '22/01/2021', 'A Man Called Magnum',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 1, 'Integer ac neque.');
```

```

insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('glissimore0@army.mil', '01/08/2023', 'Carny',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 1, 'Verdadero');
```

```

insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('glissimore0@army.mil', '01/08/2023', 'Carny',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 2, 'Falso');
```

```

insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('sshah1@deviantart.com', '24/06/2022', 'Hail Columbia!',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 1, '3');
```

```

insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('sshah1@deviantart.com', '24/06/2022', 'Hail Columbia!',
'a478fce5-8d11-45aa-a5e6-5caebb44dc27', 2, '7');
```

```

insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('sshah1@deviantart.com', '24/02/2022', 'War and Peace (Voyna i mir)',
'34fce778-900e-4b7f-b756-d217f802ccde', 1, 'ianbakfd.nañihndnbk
lrihandobi<djoifnb<ifb');
```

```

insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('sshah1@deviantart.com', '24/02/2022', 'War and Peace (Voyna i mir)',
'34fce778-900e-4b7f-b756-d217f802ccde', 2,
'sjsofinbarinafub<dvbe<bdvdlloeeeeee');
```

```

insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('aedmondson2@nationalgeographic.com', '13/06/2023', 'All About Eve',
'34fce778-900e-4b7f-b756-d217f802ccde', 1, 'Verdadero');
```

```

insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('aedmondson2@nationalgeographic.com', '13/06/2023', 'All About Eve',
'34fce778-900e-4b7f-b756-d217f802ccde', 2, 'Verdadero');
```



```
insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('aedmondson2@nationalgeographic.com', '19/07/2022', 'Cutting Edge: The
Magic of Movie Editing, The', '34fce778-900e-4b7f-b756-d217f802ccde', 1,
'5');
```

```
insert into rispostedegliutenti (Email_utente, Data_campagna,
Titolo_sondaggio, ID_organizzatore, NumeroDomanda, Risposta) values
('aedmondson2@nationalgeographic.com', '19/07/2022', 'Cutting Edge: The
Magic of Movie Editing, The', '34fce778-900e-4b7f-b756-d217f802ccde', 2,
'6');
```

```
CREATE TABLE IF NOT EXISTS account
```

```
(
    idOrganizzatore    varchar(50) PRIMARY KEY,
    nome                varchar(50),
    email               varchar(50),
    dataNascita         date,

    FOREIGN KEY (idOrganizzatore, email) REFERENCES organizzatore (id,
email),
    UNIQUE(nome)

);
```

```
INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('a478fce5-8d11-45aa-a5e6-5cae44dc27','Luettgen-Franecki','cfinlaison0@c
nn.com','23/04/1500');
```

```
INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('34fce778-900e-4b7f-b756-d217f802ccde','Zulauf
LLC','jlimer1@xinhuanet.com','04/05/2020');
```

```
INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('056acffa-a285-4890-9c25-4ce123fbbf35',
'Joscelin','jjahnke2@eventbrite.com','26/06/2003');
```

```
INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('0c3b1f6c-6878-4224-a91c-3952b2fc7775',
'Jewell','jburdess3@chronoengine.com','30/07/2000');
```

```

INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('29926989-171a-484f-94bc-84eb2f549c4a', 'Claribel','cmidner4@digg.com',
'03/03/1003');
INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('c16d45e5-92de-452a-bd3a-811c64649158', 'Carlos', 'cthirtle5@w3.org',
'20/03/2005');
INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('7af28b7c-e7d8-48fc-b3ba-25fc88615c8e', 'Mayer
Inc', 'lbernet6@mysql.com', '12/12/2012');
INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('0df773f4-b97e-4337-981b-d850268b1f1c', 'Robel-Pagac', 'gdummett7@ovh.net',
'03/03/2003');
INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('97d99bef-4e1d-4aa0-81e0-f81d341efdd6', 'Dory', 'dpartner8@imageshack.us',
'04/04/2004');
INSERT INTO account(idOrganizzatore,nome,email,dataNascita) VALUES
('a6b2a2ba-0029-45d3-a756-94c6b31a5ff7', 'Kerrill', 'klanigan9@yelp.com',
'20/03/2004');

```

```

CREATE OR REPLACE FUNCTION public.check_campagna_after_sondaggio()
    RETURNS trigger
    LANGUAGE 'plpgsql'
AS $BODY$
    BEGIN
        IF(exists( select *
                    from campagna as c join sondaggio as s on
                    (c.titolo_sondaggio,c.id_account) =
(s.titolo,s.id_organizzatore)
                    where data < data_inizio))
            THEN
                RAISE exception 'Campagna cant go before sondaggio';
            END IF;

        RETURN NEW;
    END
$BODY$;

```

```

create or replace trigger campagna_after_sondaggio
after insert on campagna
execute procedure public.check_campagna_after_sondaggio();

```

```

CREATE OR REPLACE FUNCTION public.check_organizzatore_has_account()

```

```

    RETURNS trigger
    LANGUAGE 'plpgsql'
AS $BODY$
    BEGIN
        IF(not exists(      Select * from account as a where a.idorganizzatore
= new.id_organizzatore))
        THEN
            RAISE exception 'The organizer doesnt have an account';
        END IF;

        RETURN NEW;
    END
$BODY$;

create or replace trigger organizzatore_has_account
before insert on sondaggio
for each row
execute procedure public.check_organizzatore_has_account();

CREATE OR REPLACE FUNCTION public.insert_new_theme()
    RETURNS trigger
    LANGUAGE 'plpgsql'
AS $BODY$
    BEGIN
        IF(not exists(Select * from tema as t where t.nome = new.nome_tema))
        THEN
            insert into tema(nome) values (new.nome_tema);
        END IF;

        RETURN NEW;
    END
$BODY$;

create or replace trigger insert_new_theme
before insert on sondaggio
for each row
execute procedure public.insert_new_theme();

CREATE OR REPLACE FUNCTION check_orario()
RETURNS TRIGGER AS $$
BEGIN
    IF (NEW.orario_inizio > NEW.orario_fine) THEN
        RAISE EXCEPTION 'L''orario di fine deve essere dopo l''orario di
inizio';

```

```

END IF;
RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE OR REPLACE TRIGGER OrarioRealizzazione
BEFORE INSERT OR UPDATE ON Realizzazione
FOR EACH ROW
EXECUTE FUNCTION check_orario();

```

## 7. Query SQL

Workpackage	Task	Responsabile
WP3	SQL: Query	Sanchez Munoz, Gabriel

### 7.1. Query con operatore di aggregazione e join: maxDurataRealizzazione

Questo Query visualizza ogni e-mail di realizzazione una volta solo con la sua durata massima, anche facciamo un join con alcuni dei suoi identificatori esterni, in questo caso con campagna e con utente

```

SELECT realizzazione.email, MAX(realizzazione.durata) AS maxDurata
FROM realizzazione
JOIN campagna ON realizzazione.data_campagna = campagna.data
JOIN utente ON realizzazione.email = utente.email
GROUP BY realizzazione.email

```

### 7.2. Query nidificata complessa: conteggioDomandeInfMedia

Questa query conta il numero di domande presenti nei sondaggi, ma solo quelle il cui tempo medio di completamento è inferiore al tempo medio di completamento di tutti i sondaggi.

```

select titolosondaggio, count(*)
from (Select numerodomanda, titolosondaggio,idorganizzatore
      from domandeaaperti
      where (titolosondaggio,idorganizzatore) in
            (select titolo_sondaggio as titolosondaggio,id_organizzatore as
idorganizzatore
            from ( select titolo_sondaggio, id_organizzatore, avg(durata) as durataMedia
                  from realizzazione
                  group by (titolo_sondaggio,id_organizzatore))
            where durataMedia < (select avg(durata)
                                from realizzazione))
      union all

```

```

Select numerodomanda, titolosondaggio,idorganizzatore
from domandesceltemultiple
where (titolosondaggio,idorganizzatore) in
      (select titolo_sondaggio as titolosondaggio,id_organizzatore as
idorganizzatore
      from ( select titolo_sondaggio, id_organizzatore, avg(durata) as durataMedia
            from realizzazione
            group by (titolo_sondaggio,id_organizzatore))
      where durataMedia < (select avg(durata)
                           from realizzazione))
union all
Select numerodomanda, titolosondaggio,idorganizzatore
from scalavotodomande
where (titolosondaggio,idorganizzatore) in
      (select titolo_sondaggio as titolosondaggio,id_organizzatore as
idorganizzatore
      from ( select titolo_sondaggio, id_organizzatore, avg(durata) as durataMedia
            from realizzazione
            group by (titolo_sondaggio,id_organizzatore))
      where durataMedia < (select avg(durata)
                           from realizzazione))
)
group by (titolosondaggio)
```

### 7.3. Query insiemistica:

*Questa Query mostra gli utenti che hanno risposto agli sondaggi con titolo 'Carny' oppure 'Hail Columbia!'*

```

SELECT u.nome, u.email
FROM utente u JOIN(
    SELECT r.email_utente
    FROM rispostedegliutenti r, Sondaggio s
    WHERE r.titolo_sondaggio = s.titolo AND s.titolo = 'Carny')
AS TEMP ON u.email = TEMP.email_utente
UNION
SELECT u.nome, u.email
FROM utente u JOIN(
    SELECT r1.email_utente
    FROM rispostedegliutenti r1, Sondaggio s1
    WHERE r1.titolo_sondaggio = s1.titolo AND s1.titolo = 'Hail Columbia!')
AS TEMP1 ON u.email = TEMP1.email_utente
```

## 8. Viste

<b>Workpackage</b>	<b>Task</b>	<b>Responsabile</b>
<b>WP4</b>	Viste	Corzo Galdo, Jaime

### 8.1. Viste numerodicampagne,numerodiutenti,numeroDomande

Queste viste servono per identificare il numero di campagne di un sondaggio, utenti che hanno risposto un sondaggio, e il numero di domande di un sondaggio.

I primi due sono triviali, ma nella vista numerodomande è necessario fare le unione degli tre tipi di domande.

Unset

```
CREATE VIEW numerodicampagne
AS
SELECT titolo_sondaggio, id_account, count(*) as numeroDiCampagne
FROM campagna
GROUP BY (titolo_sondaggio, id_account);
```

```
CREATE VIEW numerodiutenti
AS
SELECT titolo_sondaggio,id_organizzatore, count(*) as numeroUtenti
FROM realizzazione
Group by (titolo_sondaggio,id_organizzatore);
```

```
CREATE VIEW numeroDomande
AS
SELECT titolosondaggio,idorganizzatore, count(*) as numeroDomande
FROM
(
    SELECT titolosondaggio,idorganizzatore,numerodomanda
    FROM domandeaaperti
    UNION ALL
    SELECT titolosondaggio,idorganizzatore,numerodomanda
    FROM domandesceltemultiple
    UNION ALL
    SELECT titolosondaggio,idorganizzatore,numerodomanda
    FROM scalavotodomande
)
group by (titolosondaggio,idorganizzatore);
```

### 8.1.1. Query con Vista: Titolo query

Questa query è un esempio di un report di un sondaggio dove si possono vedere diverse informazioni non presenti direttamente nelle tabelle. Si usano le viste di sopra per ottenere questa informazione.

Unset

```
SELECT titolosondaggio,numerodomande,numeroUtenti,numeroDiCampagne,
avgTime, data
FROM numeroDomande
    join numerodiutenti on
        (numeroDomande.titolosondaggio,numeroDomande.idorganizzatore)
=
    (numerodiutenti.titolo_sondaggio,
numerodiutenti.id_organizzatore)
    join numerodicampagne on
        (numerodiutenti.titolo_sondaggio,
numerodiutenti.id_organizzatore) =
        (numerodicampagne.titolo_sondaggio,
numerodicampagne.id_account)
    join (
        select titolo_sondaggio as titolo,id_organizzatore
as idOrg,avg(durata) as avgTime
        from realizzazione
        GROUP BY (titolo_sondaggio,id_organizzatore)
    ) on
        (numerodiutenti.titolo_sondaggio,
numerodiutenti.id_organizzatore) =
        (titolo, idOrg)
    join report on
        (titolo, idOrg) =
        (report.titolo_sondaggio,report.id_organizzatore)
where titolosondaggio in (SELECT titolo_sondaggio FROM report);
```

## 9. Trigger

### 9.1. Trigger inizializzazione:

Workpackage	Task	Responsabile
WP1	Trigger inizializzazione/popoloamento database	Sanchez Munoz, Gabriel

*TRIGGER1: campagna\_after\_sondaggio*

*Questo trigger controlla che ogni campagna relativa a un sondaggio sia creata dopo la creazione del sondaggio*

Unset

```
CREATE OR REPLACE FUNCTION public.check_campagna_after_sondaggio()
  RETURNS trigger
  LANGUAGE 'plpgsql'
AS $BODY$
  BEGIN
    IF(exists( select *
              from campagna as c join sondaggio as s on
              (c.titolo_sondaggio,c.id_account) =
              (s.titolo,s.id_organizzatore)
              where data < data_inizio))
    THEN
      RAISE exception 'Campagna cant go before sondaggio';
    END IF;

    RETURN NEW;
  END
$BODY$;
create or replace trigger campagna_after_sondaggio
before insert on campagna
execute procedure public.check_campagna_after_sondaggio();
```

*TRIGGER2: OrarioRealizzazione*

*Questo trigger controlla che l'orario di inizio della realizzazione di un sondaggio sia prima che l'orario di fine*



```

Unset
-- Establecer la configuración del estilo de fecha si es necesario
-- set datestyle to european;

-- Crear el disparador (trigger)
CREATE OR REPLACE FUNCTION check_orario()
RETURNS TRIGGER AS $$
BEGIN
    IF (NEW.orario_inizio >= NEW.orario_fine) THEN
        RAISE EXCEPTION 'L'orario di fine deve essere dopo l'orario di inizio';
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE OR REPLACE TRIGGER OrarioRealizzazione
BEFORE INSERT OR UPDATE ON Realizzazione
FOR EACH ROW
EXECUTE FUNCTION check_orario();

```

**TRIGGER3: dataSondaggio**

*Questo trigger controlla che la data di fine di un sondaggio sia dopo la data di inizio del sondaggio*

```

Unset
-- Establecer la configuración del estilo de fecha si es necesario
-- set datestyle to european;

-- Crear el disparador (trigger)
CREATE OR REPLACE FUNCTION check_data()
RETURNS TRIGGER AS $$
BEGIN
    IF (NEW.data_inizio >= NEW.data_fine) THEN
        RAISE EXCEPTION 'La data di fine del sondaggio deve essere dopo la data di
inizio';
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE OR REPLACE TRIGGER dataSondaggio
BEFORE INSERT OR UPDATE ON sondaggio
FOR EACH ROW
EXECUTE FUNCTION check_data();

```

## 9.2. Trigger per vincoli aziendali

<i>Workpackage</i>	<i>Task</i>	<i>Responsabile</i>
<b>WP4</b>	Trigger per vincoli aziendali	Corzo Galdo, Jaime

### 9.2.1. Trigger1: organizzatore\_has\_account

Questo trigger controlla la RV6: un organizzatore non può creare un sondaggio senza avere un'account creato

```
Unset
set datestyle to European;

CREATE OR REPLACE FUNCTION public.check_organizzatore_has_account()
  RETURNS trigger
  LANGUAGE 'plpgsql'
AS $BODY$
  BEGIN
    IF(not exists(      Select * from account as a where a.idorganizzatore =
new.id_organizzatore))
    THEN
      RAISE exception 'The organizer doesnt have an account';
    END IF;

    RETURN NEW;
  END
$BODY$;

create or replace trigger organizzatore_has_account
before insert on sondaggio
for each row
execute procedure public.check_organizzatore_has_account();
```

### 9.2.2. Trigger2: TitoloTrigger

Questo trigger controlla la RV10: se un organizzatore crea un sondaggio con un tema che non è nella tabella degli argomenti disponibili dobbiamo inserire il nuovo tema nell'argomento

Unset

```
set datestyle to European;
```

```
CREATE OR REPLACE FUNCTION public.insert_new_theme()  
    RETURNS trigger  
    LANGUAGE 'plpgsql'  
AS $BODY$  
    BEGIN  
        IF(not exists(      Select * from tema as t where t.nome = new.nome_tema))  
        THEN  
            insert into tema(nome) values (new.nome_tema);  
        END IF;  
  
        RETURN NEW;  
    END  
$BODY$;
```

```
create or replace trigger insert_new_theme  
before insert on sondaggio  
for each row  
execute procedure public.insert_new_theme();
```