



Università degli Studi di Napoli ‘Federico II’

FACOLTÀ DI INFORMATICA

Corso di Laurea Magistrale in Basi di dati e sistemi informativi I

RELAZIONE

Boh vediamo un po' che esce

Candidati:

Vincenzo Scotti

Matricola N86/1968

Federico Salzano

Relatore:

Silvio Barra

Capitolo 1

Descrizione del progetto

1.1 Descrizione e requisiti

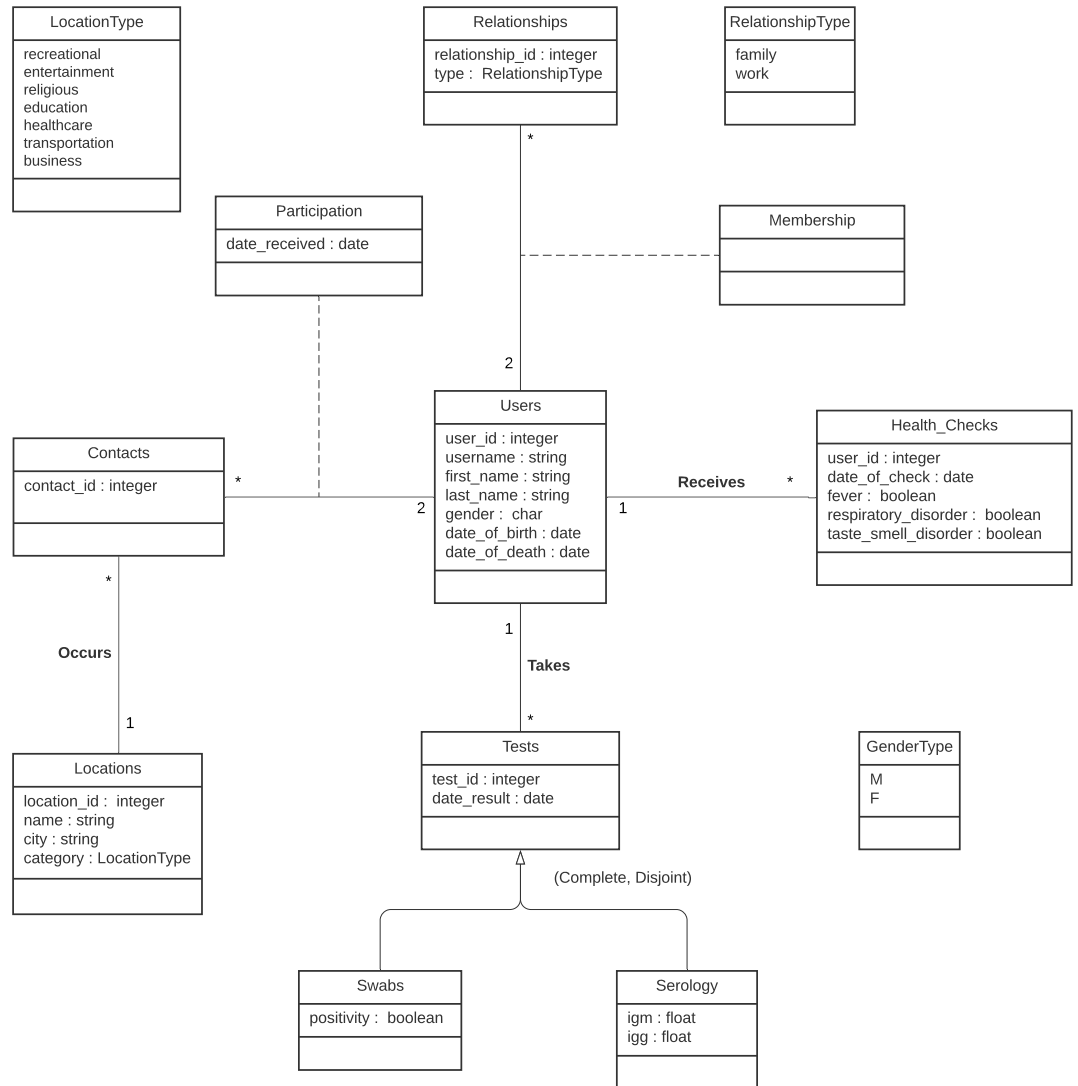
Si chiede la progettazione di una base di dati per tracciare le occasioni di contatto tra persone per recuperare informazioni sulla possibile diffusione del virus. Degli utenti va memorizzato: la possibile relazione (parentela, essere colleghi di lavoro); I luoghi in cui sono stati presenti; occasioni di contatto; tamponi e stato sierologico; informazioni sulla eventuale malattia. Si possono integrare in modo autonomo le specifiche.

Capitolo 2

Progettazione concettuale

In questo capitolo inizia la progettazione della base di dati al livello di astrazione più alto. Dal risultato dell'analisi dei requisiti che devono essere soddisfatti si arriverà ad uno schema concettuale indipendente dalla struttura dei dati e dall'implementazione fisica. In tale schema concettuale, che verrà rappresentato usando un class diagram UML, si evidenzieranno le entità (concetti) rilevanti ai fini della rappresentazione dei dati e le relazioni che intercorrono tra esse. Si delinearanno anche eventuali vincoli da imporre.

2.1 Class diagram



2.2 Ristrutturazione del class diagram

Affinchè il class diagram sia idoneo alla traduzione in schemi relazionali si deve procedere alla sua ristrutturazione, ovvero eliminare attributi strutturati, attributi multipli, generalizzazioni e specializzazioni.

2.2.1 Rimozione delle specializzazioni

La specializzazione di **Tests** in **Swabs** e **Serological tests**, essendo di tipo totale e disgiunta, verrà eliminata "schiacciando" la superclasse nelle sottoclassi.

2.2.2 Valutazione ulteriore

Procedendo con la costruzione del database ci siamo resi conto delle varie problematiche che sorgevano a causa di associazioni simmetriche e per ovviare a questo problema abbiamo strutturato il tutto in modo che sia più efficiente sia in termini di spazio che di velocità di ricerca, il risultato finale è il seguente:

2.3 Dizionario dei dati

2.3.1 Dizionario delle classi

Classe	Descrizione	Attributi
User	Descrittore di ciascun utente presente nella collezione di dati.	User_id (string): chiave sintetica per identificare univocamente un user. Username (string): First_Name (string): Nome dell'utente. Last_name (string): Cognome dell'utente. Gender (char): sesso dell'utente. Date_of_birth (date): data di nascita dell'utente. Date_of_death (date): possibile data di morte dell'utente.
Swab	Descrittore di ogni possibile tampone a cui un utente si è sottoposto.	Swab_id (integer): chiave sintetica per identificare univocamente uno swab. Date_result (date): data in cui si è ottenuto il risultato del tampone. Positivity (char): eventuale positività o negatività del tampone.

Serological_test	Descrittore di ogni possibile test sierologico a cui un utente si è sottoposto.	Serological_test_id (integer): chiave sintetica per identificare univocamente un test sierologico. Date_result (date): data in cui si è ottenuto il risultato del test. IGM (char): eventuale presenza di anticorpi IGM. IGG (char): eventuale presenza di anticorpi IGG.
Health_check	Descrittore del quadro clinico di ogni utente per data di controllo.	Health_check_id (integer): chiave sintetica per identificare univocamente ogni quadro clinico di un utente. Date_of_check (date): data in cui è avvenuto il controllo sullo stato di salute dell'utente. Fever (char): eventuale presenza del sintomo della febbre. Respiratory_disorder (char): eventuale presenza del sintomo della mancanza di fiato. Smell_taste_disorder (char): eventuale presenza del sintomo della perdita del senso del gusto e dell'olfatto.
Membership	Descrittore dell'appartenenza ad una relazione.	User_id (integer): chiave esterna per identificare a quale user ci si riferisce. Relationship_id (integer): chiave esterna per identificare a quale relationship ci si riferisce.
Participant	Descrittore dell'appartenenza ad un contatto.	User_id (integer): chiave esterna per identificare a quale user ci si riferisce. Contact_id (integer): chiave esterna per identificare a quale contact ci si riferisce.

Relationship	Descrittore di una relazione fra due user.	Relationship_id (integer): chiave sintetica per identificare univocamente la relazione. type (string): descrittore del tipo di relazione tra utenti.
Membership	Descrittore dell'appartenenza ad una relazione.	User_id (integer): chiave esterna per identificare a quale user ci si riferisce. Relationship_id (integer): chiave esterna per identificare a quale relationship ci si riferisce.
Contact	Descrittore dell'appartenenza ad una relazione.	Contact_id (integer): chiave sintetica per identificare univocamente ogni contatto. Location_id (integer): chiave esterna per identificare a quale location ci si riferisce.
Location	Descrittore dell'appartenenza ad una relazione.	Location_id (integer): chiave sintetica per identificare univocamente ogni location. Name (string): Descrittore del nome del luogo. City (string): Descrittore della città in cui è presente il luogo. Category (string): Descrittore del tipo di luogo.

2.3.2 Dizionario delle associazioni

Nome	Descrizione	Classi Coinvolte
Takes	Esprime l'aver fatto un tampone o un test serologico.	User [1] : indica l'utente che ha effettuato uno dei due test Swab (Serological_test)[*] : indica i test effettuati dagli utenti.
Receives	Esprime l'aver ricevuto un quadro clinico.	User [1] : indica l'utente che ha ricevuto il quadro clinico

		Healt_check[*] : indica i quadri clinici ricevuti dagli utenti.
Takes	Esprime l'aver fatto un tampone o un test serologico.	User [1] : indica l'utente che ha effettuato uno dei due test Swab (Serological_test)[*] : indica i test effettuati dagli utenti.
Occurs	Esprime l'occorenza di un determinato contatto in un determinato luogo.	Location [1] : indica il luogo in cui è avvenuto il contatto Swab (Contact)[*] : indica i contatti avvenuti.

Capitolo 3

Progettazione logica

In questo capitolo sarà trattata la fase successiva della progettazione della base di dati scendendo ad un livello di astrazione più basso rispetto alla precedente. Si tradurrà lo schema concettuale (già predisposto in seguito alla ristrutturazione) in uno schema logico, dipendente dal tipo di struttura dei dati prescelto cioè quello relazionale puro. Negli schemi relazionali che seguiranno le chiavi primarie sono indicate con una singola sottolineatura mentre le chiavi esterne con una doppia sottolineatura.

3.1 Traduzione in schemi relazionali

USER (User_id, Username, First_name, Last_name, Gender, Place_of_birth, Date_of_birth, Date_of_death)

Chiavi esterne: nessuna.

SWAB (Swab_id, User_id, Date_result, Positivity)

Chiavi esterne: User_id \twoheadrightarrow USER.User_id.

SEROLOGICAL_TEST (Serological_test_id, User_id, Date_result, IGM, IGG)

Chiavi esterne: User_id \twoheadrightarrow USER.User_id.

HEALTH_CHECK (Health_check_id, User_id, Date_of_check, Fever, Respiratory_disorder, Smell_taste_disorder)

Chiavi esterne: User_id \twoheadrightarrow USER.User_id.

MEMBERSHIP (User_id, Relationship_id)

Chiavi esterne: User_id \twoheadrightarrow USER.User_id;
Relationship_id \twoheadrightarrow RELATIONSHIP.Relationship_id.

PARTICIPANT (User_id, Contact_id)

Chiavi esterne: User_id \twoheadrightarrow USER.User_id;
Contact_id \twoheadrightarrow CONTACT.Contact.Contact_id.

RELATIONSHIP (Relationship_id, Type)

Chiavi esterne: nessuna.

CONTACT (Contact_id, Location_id)

Chiavi esterne: Location_id \longrightarrow LOCATION.Location_id.