

ANALISI DEL PROGETTO

Struttura logica del database

Il nucleo centrale del servizio realizzato risiede all'interno del database. Esso infatti contiene tutti i dati relativi agli studenti, al questionario, e alle risposte date da ciascuno studente.

Il modello scelto per la realizzazione è di tipo relazionale. Si procede ad illustrare la sua struttura logica, attraverso l'utilizzo di un diagramma Entity-Relationship:

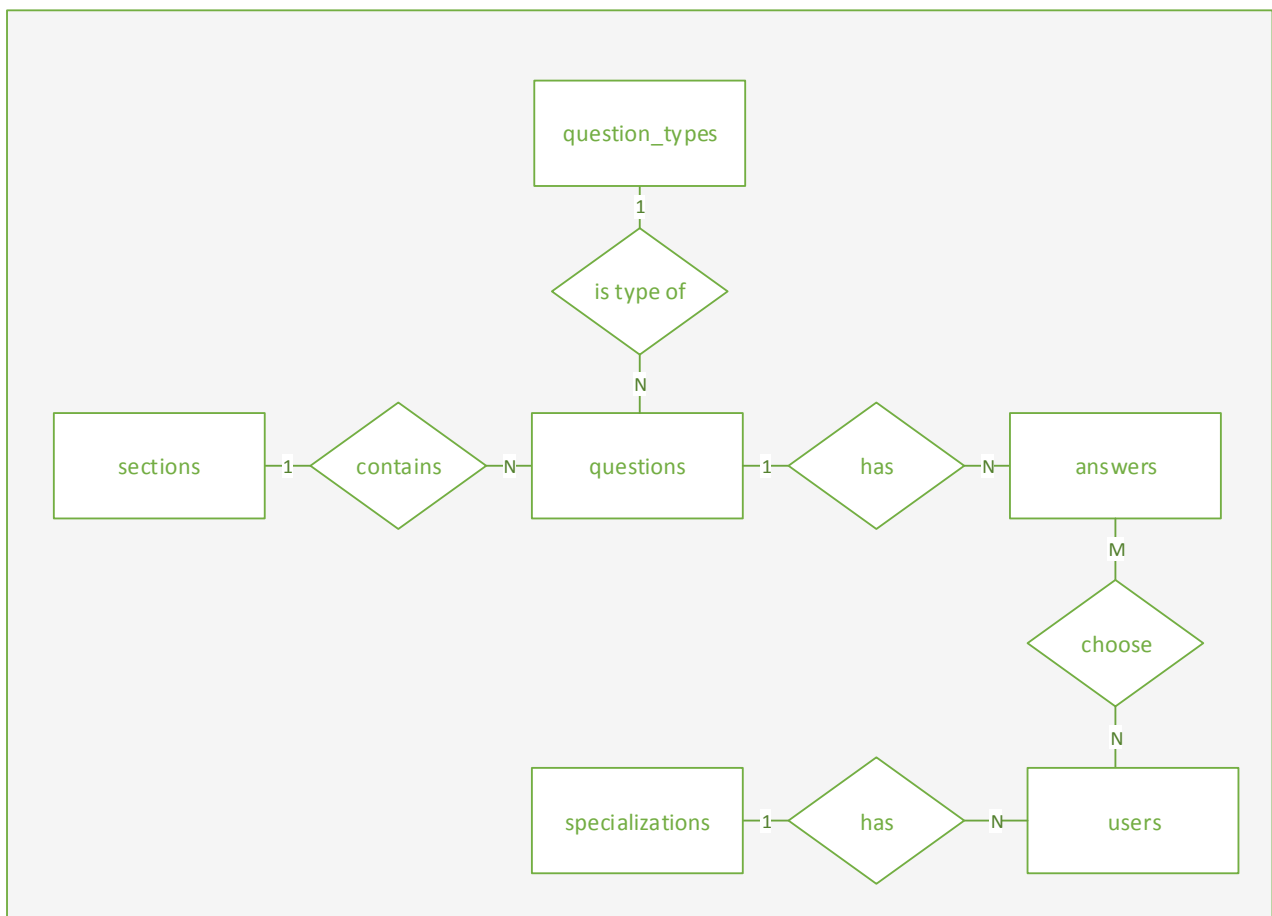


Figura 1 – Diagramma Entity-Relationship del modello scelto

ANALISI ENTITÀ

Le entità identificate nella modellazione del database sono le seguenti:

- **sections** – contiene un elenco delle sezioni (gruppi di domande relative ad un argomento comune)
- **questions** – contiene un elenco delle domande presenti all'interno del questionario
- **question_types** – contiene un elenco delle varie tipologie di domande presentate
- **answers** – contiene un elenco di tutte le possibili risposte per ciascuna domanda
- **users** – contiene un elenco degli studenti a cui verrà presentato il dizionario
- **specializations** – contiene un elenco delle singole specializzazioni di cui gli studenti hanno fatto parte

ANALISI RELAZIONI TRA LE ENTITÀ

Le relazioni presenti tra le entità identificate, sono le seguenti:

- sections - questions (1/N) – ciascuna sezione contiene una o più domande
- question_types - questions (1/N) – ciascuna domanda appartiene ad una delle tipologie stabilite
- questions - answers (1/N) – a ciascuna domanda sono associate un certo numero di risposte
- answers - users (M/N) – un singolo studente può scegliere più risposte all'interno del questionario, ma allo stesso modo una singola risposta può essere scelta da più studenti distinti
- specializations - users (1/N) – ciascuno studente è stato iscritto ad una determinata specializzazione durante il suo corso di studi

Struttura fisica del database

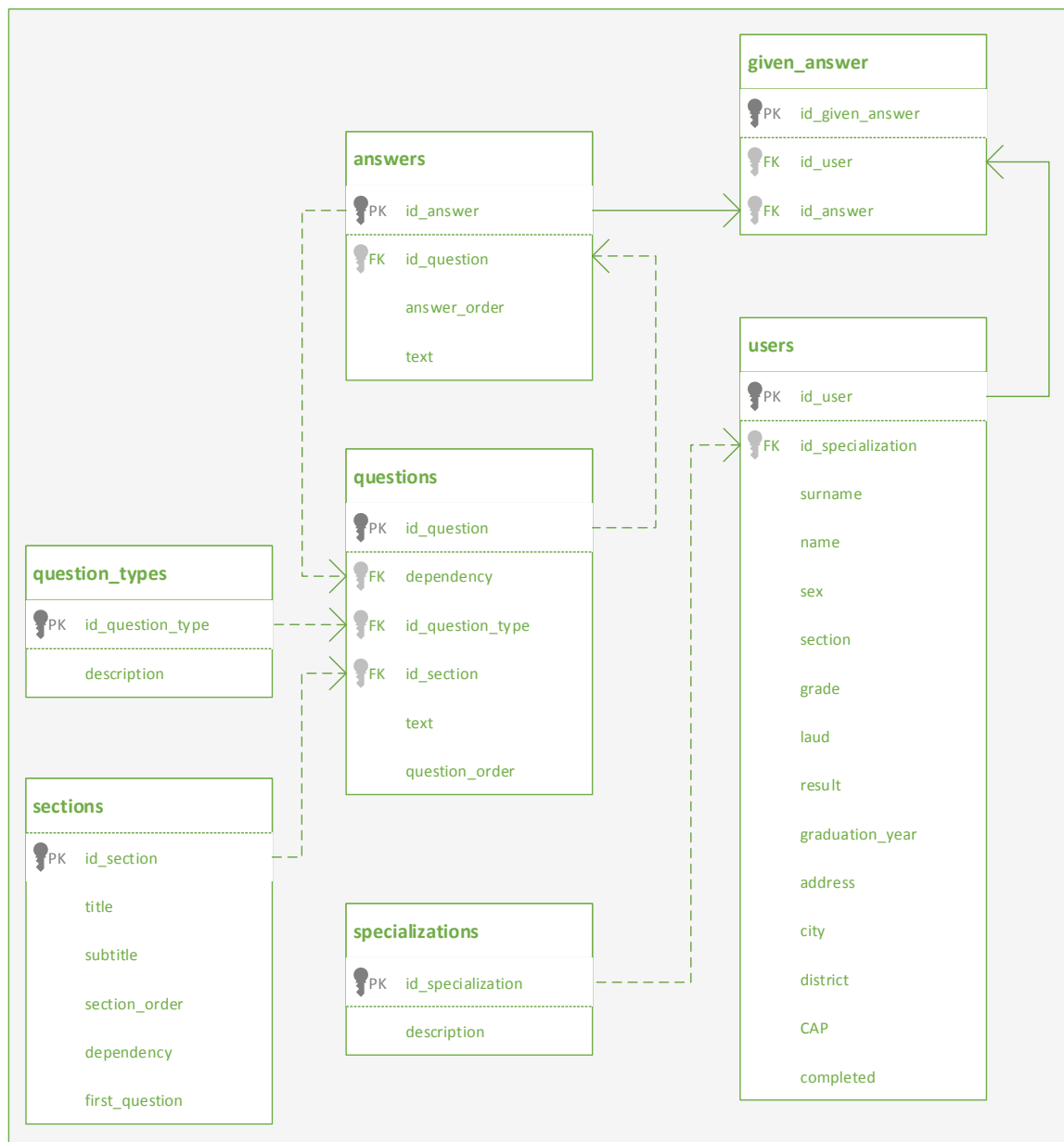


Figura 2 – Diagramma UML della struttura fisica del database

Si illustra in figura 2 la struttura fisica scelta per l'implementazione della base di dati, realizzata mediante l'RDBMS (Relational Database Management System) MySQL.

LA TABELLA GIVEN_ANSWERS

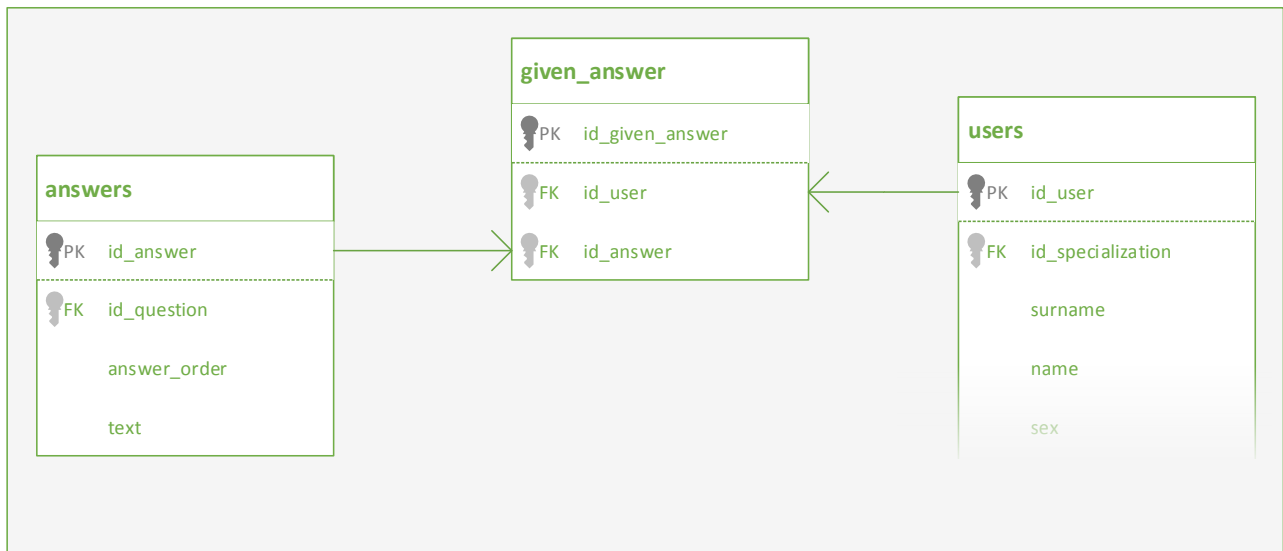


Figura 3 – Particolare del diagramma UML della struttura del database, che mostra la tabella given_answer

Un limite dei database relazionali consiste nella loro impossibilità di realizzare relazioni many-to-many (M:N), come quella presente tra le entità **users** e **answers**. Tale limite è però facilmente superabile mediante l'introduzione di una tabella di giunzione (junction table), e a due relazioni di tipo one-to-many (1:N) dalle due tabelle coinvolte a quest'ultima.

La junction table realizzata assume il nome **given_answer**, e contiene un elenco delle risposte date da ciascun utente. E' opportuno notare come tecnica permetta la semplice risoluzione di problemi altrimenti ostici, come la scelta di risposte multiple per una singola domanda (è sufficiente inserire più entry, una relativa ad ogni risposta).

FLESSIBILITÀ DELL MODELLO UTILIZZATO

Una delle maggiori sfide presenti all'interno del progetto è la realizzazione di una base di dati flessibile, capace di adattarsi in modo rapido e semplice ad eventuali cambiamenti. La struttura proposta segue tale principio, in quanto ogni singola informazione relativa al questionario è compresa all'interno del database.

Aggiungere una nuova domanda, cambiarne la tipologia o l'ordine, correggere il testo di una risposta: sono tutte operazioni effettuabili mediante semplici modifiche all'interno del database. Non è richiesta alcun tipo di modifica al codice dell'applicazione.

Questo permette di ridurre gran parte dei costi di manutenzione, e limita l'eventualità di compromettere il sistema nel caso siano necessarie modifiche.

PROBLEMATICHE DELLE DIPENDENZE DOMANDA – RISPOSTA

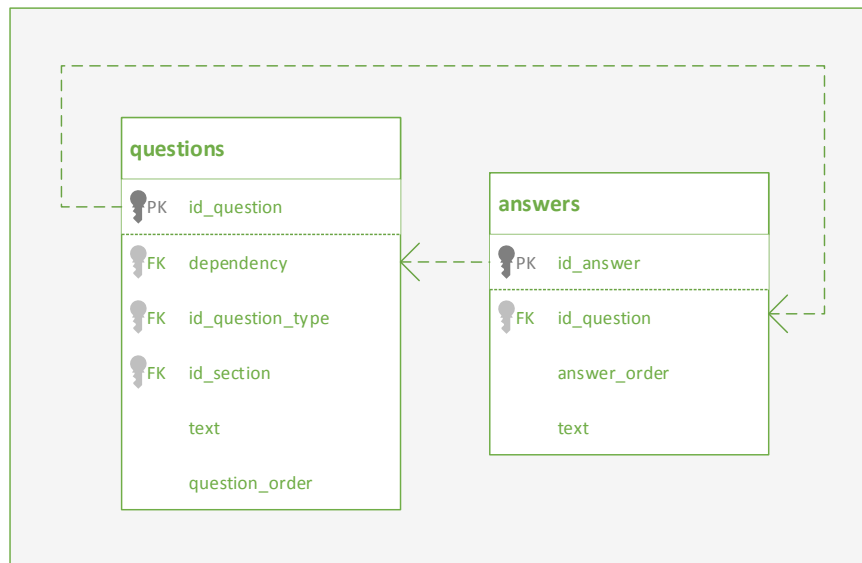


Figura 4 – Particolare del diagramma UML della struttura del database, che mostra le tabelle questions e answers

Nell'ottica di una completa separazione tra dati e applicativo, risulta necessario introdurre un meccanismo per modellare le dipendenze presenti tra determinate domande ed altre risposte. Si consideri, per esempio, come ad uno studente che ha dichiarato di non aver mai frequentato l'università, non sia necessario effettuare domande relative al corso di laurea scelto.

La soluzione scelta consiste nell'introduzione di un campo opzionale dependency (dipendenza) all'interno della tabella questions. Se la risposta identificata da tale campo è stata selezionata in precedenza la relativa domanda viene mostrata. Se tale risposta non è stata selezionata, la domanda invece non viene mostrata.

NORMALIZZAZIONE

Nell'ottica dell'eliminazione di ridondanze informative, e conseguentemente del rischio introdurre uno stato di incoerenza all'interno del database, sono stati seguiti una serie di accorgimenti volti alla riduzione dello stesso alla terza forma normale.

La normalizzazione consiste di fatto nella riduzione di tabelle che presentano campi interdipendenti in tabelle più piccole. Nel caso sia necessario ottenere la tabella originaria per effettuare un'interrogazione è possibile utilizzare un'operazione di join.

Tecnologie utilizzate

Si effettua adesso una rassegna delle tecnologie impiegate. In una successiva sezione si avrà modo di analizzare nello specifico l'utilizzo di ciascuna tecnologia all'interno del progetto Outlook.

PHP, MYSQL E MYSQLI

Una tecnologia fondamentale ai fini della realizzazione dell'applicativo lato server è PHP.

PHP, acronimo di "PHP: Hypertext Preprocessor", è un linguaggio di programmazione interpretato nato per la programmazione web. Al giorno d'oggi è largamente utilizzato per sviluppare applicazioni *lato server*.

All'interno dell'applicativo PHP (nello specifico la versione 5.2.17) viene utilizzato per la creazione di pagine dinamiche, come quella di presentazione del questionario, e per la realizzazione dei meccanismi di login e di invio delle risposte.

L'interfaccia di collegamento tra l'interprete PHP e l'RDBMS MySQL è fornita dalla libreria MySQLi (MySQL Improved). Tale libreria consiste essenzialmente in un miglioramento della precedente libreria MySQL: permette un approccio di tipo orientato agli oggetti (mantenendo intatta la possibilità di sfruttare un approccio procedurale), introduce l'utilizzo dei prepared statements (fondamentali per eliminare il rischio di attacchi di tipo SQL Injection) e permette la realizzazione di tecniche quali le transazioni.

HTML5 E CSS3

La realizzazione del sito web collegato al servizio Outlook, è stata realizzata mediante l'utilizzo dei linguaggi di markup HTML5 e CSS3.

HTML5 (Hyper Text Transfer Protocol) è comunemente utilizzato per la realizzazione di pagine web.

CSS3 (Cascading Style Sheets) è usato per definire la formattazione e lo stile di documenti HTML.

L'utilizzo di tali tecnologie permette una completa separazione tra il significato semantico del contenuto e l'aspetto grafico, relativo al modo in cui i contenuti sono presentati. I vantaggi sono una maggiore chiarezza e la possibilità di riutilizzare parti di codice.

JAVASCRIPT E JQUERY

La natura dinamica del sito web realizzata, è possibile grazie all'utilizzo di tecnologie quali il Javascript, e di una sua potente libreria, jQuery.

JavaScript è un linguaggio di scripting orientato agli oggetti comunemente usato nella creazione di siti web, lato client-side. Tutti i moderni browser web, infatti, dispongono di un interprete Javascript.

jQuery è una potente libreria Javascript che si propone lo scopo di semplificare la programmazione lato client delle pagine HTML. Per farlo, fornisce una serie di strumenti che permettono di effettuare svariate operazioni con poche e semplici linee di codice.

L'utilizzo di queste tecnologie ha trovato uso, specificatamente, nella realizzazione del meccanismo di visualizzazione di domande dipendenti da determinate risposte.

Parti di codice php (submit delle risposte)

Parti di codice js (mostra/nascondi domande)

Considerazioni sulla sicurezza

TRANSAZIONI

PREPARED STATEMENTS

"SICUREZZA DELLE PASSWORD"

Ulteriori considerazioni

REALIZZAZIONE DNS

CONCLUSIONI