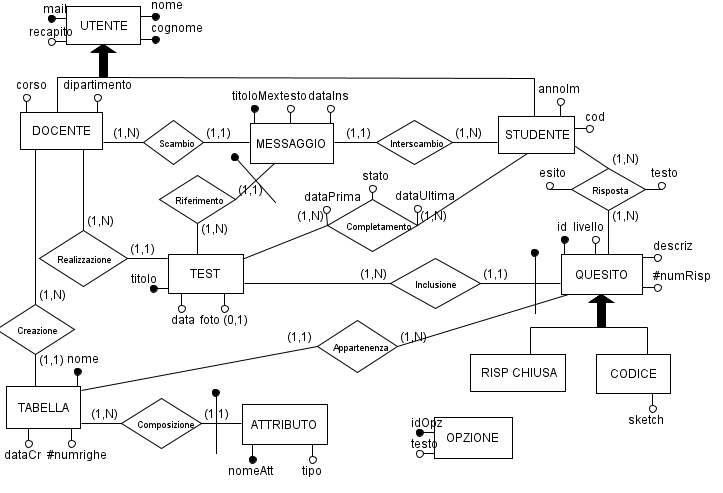
**RELAZIONE**

**1) Raccolta/Analisi dei requisiti** (testo completo delle specifiche sui dati, lista delle operazioni, tavola media dei volumi, glossario dei dati).

1. Definizione di un documento di specifica. 2. Decomposizione del testo in gruppi di frasi. 3. Costruzione di un glossario. 4. Definizione delle operazioni sui dati.

**2) Progettazione Concettuale** (diagramma E-R, dizionario delle entità/relazioni, tavola delle business rules).

****

**Dizionario delle Entità:**

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITA’ | DESCRIZIONE | ATTRIBUTI | IDENTIDIFICATORE |
| Utente | Linea ferroviaria su cui circolano i treni | mail, nome, cognome, recapito | mail, nome, cognome |
| Docente | Parte di una linea ferroviaria | " + corso, dipartimento | " |
| Studente | Luogo dove si trovano linee e treni | " + annoIm, cod | " |
| Tabella | Mezzo di trasporto | nome, dataCr, #numrighe | nome |
| Attributo | Treno dedito al solo trasporto di merci | nomeAtt, tipo | nomeAtt + Identif. Esterno |
| Test | Treno dedito al solo trasporto di persone | titolo, data, foto | titolo |
| Quesito | Azienda ferroviaria | id, livello, descr, #numRisp | Id + Identif. Esterno |
| Risp. Chiusa | Operazione di manutenzione |  | " |
| Codice | Intervento che si ripete periodicamente | " + sketch | " |
| Opzione |  |  |  |
| Risposta |  |  |  |
| Messaggio |  | titoloMex, testo, dataIns | titoloMex |

Per quanto riguarda le Generalizzazioni ho utilizzato " per indicare che le classi figlie ereditano tutti gli attributi della classe padre ed hanno lo stesso identificatore del padre. Ad esempio Treno merci e Treno passeggeri ereditano gli attributi della classe Treno. Stessa cosa per intervento periodico.

**Dizionario delle Relazioni:**

|  |  |  |  |
| --- | --- | --- | --- |
| RELAZIONI | DESCRIZIONE | COMPONENTI | ATTRIBUTI |
| Creazione | Composizione della linea. | Docente, Tabella |  |
| Realizzazione | Collegamento di stazione di partenza e stazione di arrivo | Docente, Test |  |
| Riferimento | Fermata associa una stazione ad ogni treno | Messaggio, Test |  |
| Completamento | Tracciamento della linea di ogni treno merci | Test, Studente | dataPrima, dataUltima, stato |
| Inclusione | Appartenenza di un treno ad una compagnia | Test, Quesito |  |
| Appartenenza | Un intervento fa Riferimento a quale tratta | Tabella, Quesito |  |
| Composizione | Esecuzione di un intervento da parte di una compagnia | Tabella, Attributo |  |
| Risposta |  | Studente, Quesito | esito |
| Scambio |  | Messaggio, Docente |  |
| Interscambio |  | Messaggio, Studente |  |

**Business Rules:**

|  |
| --- |
| REGOLE DI VINCOLO |
| In ogni fermata la data di arrivo deve essere sempre maggiore della data di partenza. |
|  |
| REGOLE DI DERIVAZIONE |
| Il campo #numtratte è dato dalla somma di tutte le tratte presenti in quella linea ferroviaria. |
| La lunghezza complessiva di una linea è data dalla somma delle lunghezze di ogni singola tratta. |

**3) Progettazione Logica** (ristrutturazione dello schema concettuale, analisi delle ridondanze, lista delle tabelle con vincoli di chiavi, lista dei vincoli inter-relazionali).

**Tabelle dei volumi:**

- Valutare se la seguente ridondanza: campo #numrisposte relativo ad un quesito debba essere tenuta o eliminata, sulla base delle seguenti operazioni:

* Aggiungere una nuova risposta ad un quesito esistente (10 volte/mese, interattiva)
* Rimuovere un quesito e tutte le risposte ottenute (2 volte/mese, batch)
* Visualizzare tutti gli utenti presenti nella piattaforma (1 volte/mese, batch)
* Contare il numero di risposte per ciascun quesito presente nella piattaforma (2 volte/mese, interattiva)

Coefficienti analisi: α (peso op. scrittura) =2, WI (peso op) =1, WB (peso op batch) = 0.5   
Tabella dei volumi: 10 risposte per quesito, 20 quesiti, 50 utenti

Formula: c(OT ) = ƒ(OT ) ⋅ WT ⋅ (α ⋅ NCwrite + NCread )

Caso **CON ridondanza**:

* c(Op1) = 10 ⋅ 1 ⋅ (2 ⋅ 2 + 0) = 40 -> il 2 nelle op di scrittura è dato da 1 accesso a Risposta + 1 accesso a Quesito.
* c(Op2) = 2 ⋅ 0,5 ⋅ (2 ⋅ 12 + 0) = 24 -> il 12 nelle op di scrittura è dato da 10 accesso a Risposta + 1 accesso a Quesito + 1 accesso a RispChiusa o Codice.
* c(Op3) = 1 ⋅ 0,5 ⋅ (2 ⋅ 0 + 100) = 20 -> il 100 nelle op di lettura è dato da 50 accessi a Utente + 50 accessi a Docente o Studente.
* c(Op4) = 2 ⋅ 1 ⋅ (2 ⋅ 40 + 0) = 160 -> il 40 nelle op di scrittura è dato da 20 accessi a Quesito + 20 accessi a RispChiusa o Codice.

Costo tot con ridondanza c(Srid) = 160 + 40 = 200

Caso **senza ridondanza**:

* c(Op1) = 10 ⋅ 1 ⋅ (2 ⋅ 1 + 0) = 40 -> il 1 nelle op di scrittura è dato da 1 accesso a Risposta.
* c(Op4) = 2 ⋅ 1 ⋅ (2 ⋅ 240 + 0) = 960 -> il 240 nelle op di scrittura è dato da 20 accessi a Quesito + 20 accessi a RispChiusa o Codice + 200 (20x10) accessi a Risposta.

Costo tot senza ridondanza c(S) = 960 + 40 = 1000

Per quanto riguarda l’occupazione di memoria:   
m(S) = X (byte)   
m(Srid) = X + 20 \* 4 = X + 80 (Byte) -> dove 20 rappresenta il numero di Quesiti e 4 sono i Byte.

In sintesi, la presenza della ridondanza:   
Introduce un overhead di memoria di 80 Byte e migliora lo speedup delle operazioni: 1000/200 ~ 5. In questo caso, è conveniente mantenere l’attributo #numrisposte.

**4) Normalizzazione** (se necessaria).

**5) Descrizione -ad alto livello- delle funzionalità dell’applicazione Web.**

**6) (In appendice): Codice SQL completo dello schema della base di dati.**