
Analisi e progettazione del software – Appello dell'11 luglio 2003 – Compito A

Tempo a disposizione: 150 minuti – Libri e appunti chiusi

Nome: _____

Cognome: _____

Matricola: _____

ERedit è un sistema per la rappresentazione di diagrammi Entità-Relazione (ER), che deve gestire:

- Diagrammi ER, ciascuno con un nome che lo identifica e il nome dell'autore. Ad esempio, il diagramma *Università* creato da *Mario Rossi*.
- Un diagramma ER può contenere delle entità, ciascuna con un nome (che la identifica nell'ambito di un diagramma) e una descrizione. Ad esempio, l'entità *Studente*, con descrizione *Studente di questa università*.
- Alle entità possono essere associati degli attributi, ciascuno con un nome (che lo identifica nell'ambito di una entità), un tipo e una descrizione. Ad esempio, l'attributo *Nome* di *Studente*, di tipo *String*, che è *Nome di questo studente*.
- Un diagramma ER può contenere delle relazioni binarie, ovvero, che coinvolgono esattamente due entità; sono ammesse anche relazioni ricorsive, che coinvolgono due volte la stessa entità. Ciascuna relazione ha un nome (che la identifica nel diagramma) e una descrizione. Ad esempio, l'entità *Iscrizione*, tra *Studente* e *Facoltà*, con descrizione *Lo studente è iscritto alla facoltà*.
- Anche alle relazioni possono essere associati degli attributi, ciascuno con un nome (che lo identifica nell'ambito della relazione), un tipo e una descrizione.

ERedit consente di creare e modificare diagrammi ER, mediante un certo numero di casi d'uso, nei quali sono state identificate le seguenti operazioni di sistema (oltre ad altre, che però non sono di interesse):

1. Creazione di un nuovo diagramma ER, dato il nome del diagramma e il nome dell'autore. Il diagramma è inizialmente vuoto. Da quel momento in poi, il diagramma ER creato è considerato il diagramma corrente.
 2. Inserimento di una entità nel diagramma corrente, dato il nome e la descrizione dell'entità.
 3. Inserimento di una relazione nel diagramma corrente, dato il nome e la descrizione della relazione, nonché i nomi delle due entità coinvolte dalla relazione.
 4. Inserimento di un attributo in una entità del diagramma corrente, dato nome, tipo e descrizione dell'attributo, nonché il nome dell'entità a cui va associato.
 5. Inserimento di un attributo in una relazione del diagramma corrente, dato nome, tipo e descrizione dell'attributo, nonché il nome della relazione a cui va associato.
 6. Cancellazione di una relazione dal diagramma corrente, dato il suo nome. Insieme alla relazione vanno cancellati anche tutti i suoi attributi.
 7. Cancellazione di una entità dal diagramma corrente, dato il suo nome. Insieme all'entità vanno cancellati anche tutti i suoi attributi, nonché le relazioni che coinvolgono l'entità e i loro attributi.
 8. Chiusura del diagramma corrente (che cessa di essere corrente).
-

Esercizio A1 (25%)

Fare l'analisi orientata agli oggetti relativa al sistema ERedit, relativamente alle operazioni di sistema identificate, come segue: Mostrare il modello di dominio (con riferimento a tutte le operazioni di sistema che sono state identificate). Mostrare il contratto delle operazioni di sistema 1 e 2.

Esercizio A2 (45%)

Fare la progettazione orientata agli oggetti relativa al sistema ERedit, come segue: Mostrare i diagrammi di interazione relativi alle operazioni di sistema 3, 4 e 6, motivando le scelte di progetto fatte mediante l'indicazione dei pattern GRASP applicati. Mostrare il corrispondente diagramma delle classi di progetto.
Si faccia l'ipotesi che il sistema ERedit gestisca i propri dati solo in memoria principale.

Esercizio A3 (10%) – questo esercizio va fatto DOPO e SEPARATAMENTE dall'esercizio A2

Mostrare anche i diagrammi di interazione relativi alle operazioni di sistema 5 e 7, motivando le scelte di progetto fatte mediante l'indicazione dei pattern GRASP applicati. Mostrare il nuovo diagramma delle classi di progetto.

Esercizio A4 (10%) – questo esercizio va fatto DOPO e SEPARATAMENTE dall'esercizio A2

Discutere la possibilità di introdurre le seguenti estensioni al sistema ERedit, anche facendo riferimento a design pattern opportuni:

1. Gestione di relazioni non binarie, ovvero che possono coinvolgere anche più di due entità.
 2. Gestione di attributi composti, ovvero attributi formati come composizione di altri attributi (che possono essere a loro volta semplici oppure composti).
-

Esercizio A5 (10%)

Descrivere il pattern Layers.