|  |  |
| --- | --- |
|  |  |

Basi di Dati

Progetto A.A. 2021/2022

TITOLO DEL PROGETTO

0253048

Abdelrahman Abdelrahman

**Indice**

[1. Descrizione del Minimondo 2](#_Toc606296459)

[2. Analisi dei Requisiti 4](#_Toc1289394997)

[3. Progettazione concettuale 5](#_Toc2081466291)

[4. Progettazione logica 6](#_Toc2147004904)

[5. Progettazione fisica 8](#_Toc518560220)

[Appendice: Implementazione 10](#_Toc403811585)

Tutto il testo su sfondo grigio, all’interno di questo template, deve essere eliminato prima della consegna. Viene utilizzato per fornire informazioni sulla corretta compilazione del report di progetto.

Non modificare il formato del documento:

- Carattere: Times New Roman, 12pt

- Dimensione pagina: A4

- Margini: superiore/inferiore 2,5cm, sinistro/destro: 1,9cm

L’assegnazione della tesina può essere effettuata online, visitando il sito <https://www.pellegrini.tk/progetti/> ed inserendo i propri dati. Per qualsiasi problema, contattare il docente via email all’indirizzo [a.pellegrini@ing.uniroma2.it](mailto:a.pellegrini@ing.uniroma2.it)

# Descrizione del Minimondo

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65 | Si vuole realizzare un servizio online che consenta di giocare ad un clone del famoso gioco Risiko, in modalità "conquista del mondo".  Al sistema hanno accesso due tipologie di utenti: i giocatori e i moderatori. I moderatori hanno la possibilità di creare stanze di gioco, in funzione della quantità di giocatori che utilizzano attualmente il sistema.  In particolare, i moderatori hanno la possibilità di visualizzare, tramite un report, quante stanze hanno attualmente partite in corso e quanti giocatori stanno partecipando alla partita.  Inoltre, in questo report, gli amministratori possono visualizzare il numero totale di giocatori che hanno effettuato almeno un'operazione negli ultimi 15 minuti che non sono all'interno di alcuna stanza di gioco.  Una stanza permette ad un numero massimo di sei giocatori di entrare e partecipare alla partita. Una partita coinvolge almeno tre giocatori. Quando il terzo giocatore entra in una stanza, viene attivato un countdown tale da avviare la partita dopo due minuti. In questi due minuti altri giocatori possono entrare, fino al massimo concesso.  All'avvio della partita, gli stati del tabellone vengono assegnati casualmente ai giocatori. I turni "girano" in funzione del tempo di ingresso dei giocatori nella stanza (chi è entrato prima gioca prima).  Un turno prevede che il giocatore possa compiere una delle seguenti azioni: \* Posizionare un numero arbitrario di carri armati in uno stato \* Scegliere uno stato da cui fare partire un attacco verso uno stato adiacente \* Spostare carri armati da uno stato ad un altro adiacent (almeno un carro armato deve restare nello stato di partenza)  L'attacco viene svolto nel seguente modo.  La fase di attacco si svolge tra il giocatore che attacca e quello che difende attraverso il lancio dei dadi.  Il numero dei dadi da lanciare è stabilito dal numero di armate che si decide di schierare in guerra meno una, fino ad un massimo di tre dadi per volta.  Ognuno dei giocatori lancia il numero di dadi corrispondenti, e poi si confrontano i valori ottenuti, il più alto dell'attaccante con il più alto del difensore, il secondo con il secondo e così via.  Per ogni punteggio più alto, il perdente deve togliere un'armata dal tabellone.  In caso di parità il punto va al difensore. Il lancio dei dadi viene simulato mediante la generazione di numeri pseudocasuali.  Se lo stato attaccato perde tutte le armate, questo viene conquistato e vengono spostate automaticamente in esso un numero di armate pari a quelle sopravvissute all'attacco.  Un apposito timer determina quando il tempo per svolgere un'azione da parte di un giocatore scade e il turno passa quindi al giocatore successivo.  Al termine del turno, se è stata svolta almeno un'azione, il giocatore riceve un numero di carri armati da posizionare pari al numero di stati posseduti diviso tre, arrotondato per eccesso.  Un'apposita procedura consente al client di sapere, quando è il turno del giocatore, tutto lo stato di gioco e quindi far scegliere quale azione effettuare.  Si ricorda, comunque, che tutta la logica applicativa è implementata nel DBMS.  Un giocatore può sempre visualizzare lo storico di tutte le partite giocate. |

# Analisi dei Requisiti

## Identificazione dei termini ambigui e correzioni possibili

|  |  |  |  |
| --- | --- | --- | --- |
| **Linea** | **Termine** | **Nuovo termine** | **Motivo correzione** |
| 5 | giocatori | Giocatori  Autenticati | Come i giocatori per utilizzare il sistema devono autenticarsi, altrimenti non è possibile accedere al gioco. |
| 11 | amministratore | Moderatore | Come è stato dichiarato prima come “moderatore”. |
| 13 | operazione | Azione | Come è stato dichiarato dopo che “le azioni” che il giocatore può effettuare sono. |
| 16 | stanza | Lobby  (Partita in attesa) | Come non viene definito quando viene creata la partita, ma vengono fatti i riferimenti alla stessa partita che impone che sia la stessa partita riferita che va in “Countdown” e poi viene “Avviata”.  Perciò rimane in modalità “Lobby” ospitato nella stanza di gioco relativa, prima di soddisfare i requisiti per diventare “Partita In Countdown”. |
| 17 | partita | Partita In Countdown | Come è stato definito che deve avere almeno tre giocatori per poi partecipare. |
| 9,10 | partita | Partita Avviata | Come vengono chiamate “Partite In Corso”, che implica che sono state avviate. |
|  | stato | Nazione | Come rappresenta le nazioni che esistono nel gioco e su cui si basa il tabellone e la sua assegnazione delle nazioni. |
| 22 | Stati del tabellone | Territori | Come rappresentono le istanze delle nazioni assegnate ai giocatori durante una “Partita Avviata” che nel contesto della partita. |
| 23 | giocatori | Giocatori Participanti | Come vengono definiti prima come “giocatori participanti stanno partecipando alla partita”. |
| 27,28(1),29 | stato | Territorio  Alleato | Come ogni giocatore può solamente far partire un’azione da un Territorio Alleato (Sotto il suo dominio). |
| 28(2) | stato | Territorio  Nemico | Come ogni giocatore può solamente far partire attacchi contro Territorio Nemici (Sotto il dominio di un altro giocatore). |
|  |  |  |  |

### Specifica disambiguata

|  |
| --- |
| Si vuole realizzare un servizio online che consenta di giocare ad un clone del famoso gioco Risiko, in modalità "conquista del mondo".  Al sistema hanno accesso due tipologie di utenti: i giocatori e i moderatori. I moderatori hanno la possibilità di creare stanze di gioco, in funzione della quantità di giocatori autenticati che utilizzano attualmente il sistema.  In particolare, i moderatori hanno la possibilità di visualizzare, tramite un report, quante stanze hanno attualmente partite avviate e quanti giocatori stanno partecipando alla partita.  Inoltre, in questo report, i moderatori possono visualizzare il numero totale di giocatori che hanno effettuato almeno un'operazione negli ultimi 15 minuti che non sono particpanti a nessuna partita all'interno di alcuna stanza di gioco.  Una stanza permette ad un numero massimo di sei giocatori di entrare e partecipare alla partita. Una partita coinvolge almeno tre giocatori. Quando il terzo giocatore entra in una stanza, viene attivato un countdown tale da avviare la partita dopo due minuti. In questi due minuti altri giocatori possono entrare, fino al massimo concesso.  All'avvio della partita, gli stati del tabellone vengono assegnati casualmente ai giocatori. I turni "girano" in funzione del tempo di ingresso dei giocatori nella stanza (chi è entrato prima gioca prima).  Un turno prevede che il giocatore possa compiere una delle seguenti azioni: \* Posizionare un numero arbitrario di carri armati in uno stato \* Scegliere uno stato da cui fare partire un attacco verso uno stato adiacente \* Spostare carri armati da uno stato ad un altro adiacent (almeno un carro armato deve restare nello stato di partenza) L'attacco viene svolto nel seguente modo.  La fase di attacco si svolge tra il giocatore che attacca e quello che difende attraverso il lancio dei dadi.  Il numero dei dadi da lanciare è stabilito dal numero di armate che si decide di schierare in guerra meno una, fino ad un massimo di tre dadi per volta.  Ognuno dei giocatori lancia il numero di dadi corrispondenti, e poi si confrontano i valori ottenuti, il più alto dell'attaccante con il più alto del difensore, il secondo con il secondo e così via.  Per ogni punteggio più alto, il perdente deve togliere un'armata dal tabellone.  In caso di parità il punto va al difensore. Il lancio dei dadi viene simulato mediante la generazione di numeri pseudocasuali.  Se lo stato attaccato perde tutte le armate, questo viene conquistato e vengono spostate automaticamente in esso un numero di armate pari a quelle sopravvissute all'attacco.  Un apposito timer determina quando il tempo per svolgere un'azione da parte di un giocatore scade e il turno passa quindi al giocatore successivo.  Al termine del turno, se è stata svolta almeno un'azione, il giocatore riceve un numero di carri armati da posizionare pari al numero di stati posseduti diviso tre, arrotondato per eccesso.  Un'apposita procedura consente al client di sapere, quando è il turno del giocatore, tutto lo stato di gioco e quindi far scegliere quale azione effettuare. Si ricorda, comunque, che tutta la logica applicativa è implementata nel DBMS. Un giocatore può sempre visualizzare lo storico di tutte le partite giocate. |

## Glossario dei Termini

Realizzare un dizionario dei termini, compilando la tabella qui sotto, a partire dalle specifiche precedentemente disambiguate

|  |  |  |  |
| --- | --- | --- | --- |
| **Termine** | **Descrizione** | **Sinonimi** | **Collegamenti** |
|  |  |  |  |

## Raggruppamento dei requisiti in insiemi omogenei

Per ciascun elemento “più importante” della specifica (riportata anche nel glossario precedente), estrapolare dalla specifica disambiguata le frasi ad esso associate. Compilare una tabella separata per ciascun elemento individuato.

|  |
| --- |
| **Frasi relative a ...** |
|  |

# Progettazione concettuale

## Costruzione dello schema E-R

In questa sezione è necessario riportare tutti passi seguiti per la costruzione dello schema E-R finale, a partire dalle specifiche raccolte ed organizzate nel capitolo precedente. Non è richiesto un procedimento specifico: si può adottare una strategia top-down, bottom-up, a macchia d’olio o mista. L’importante è descrivere e commentare tutti i passi della costruzione, andando anche ad inserire “schemi parziali” utilizzati nel processo.

### Integrazione finale

Nell’integrazione finale delle varie parti dello schema E-R è possibile che si evidenzino dei conflitti sui nomi utilizzati e dei conflitti struttuali. Prima di riportare lo schema E-R finale, descrivere quali passi sono stati adottati per risolvere tali conflitti.

## Regole aziendali

Laddove la specifica non sia catturata in maniera completa dallo schema E-R, corredare lo stesso in questo paragrafo con l’insieme delle regole aziendali necessarie a completare la progettazione concettuale.

## Dizionario dei dati

Completare la progettazione concettuale riportando nella tabella seguente il dizionario dei dati

|  |  |  |  |
| --- | --- | --- | --- |
| **Entità** | **Descrizione** | **Attributi** | **Identificatori** |
|  |  |  |  |

# Progettazione logica

## Volume dei dati

Questa sezione serve ad illustrare qual è il carico che la base di dati dovrà sopportare. A tal fine, è necessario prevedere un volume di dati attesi. Compilare la tabella sottostante, per ciasun concetto identificato nello schema E-R. I volumi devono essere stimati dallo studente in maniera ragionevole rispetto all’operatività presunta dell’applicativo.

|  |  |  |
| --- | --- | --- |
| **Concetto nello schema** | **Tipo[[1]](#footnote-0)** | **Volume atteso** |
|  |  |  |

## Tavola delle operazioni

Rappresentare nella tabella sottostante tutte le operazioni sulla base di dati che devono essere supportate dall’applicazione, con la frequenza attesa. Le operazioni da supportare devono essere desunte dalle specifiche raccolte.

|  |  |  |
| --- | --- | --- |
| **Cod.** | **Descrizione** | **Frequenza attesa** |
|  |  |  |

## Costo delle operazioni

In riferimento a tutte le operazioni precedentemente indicate, calcolarne il costo supponendo, per questa fase del progetto, che il costo in scrittura di un dato sia doppio rispetto a quello in lettura.

## Ristrutturazione dello schema E-R

Descrivere (laddove necessario fornendo anche degli schemi) quali passi vengono adottati per ristrutturare lo schema E-R, ad esempio in termini di:

Analisi delle ridondanze

Eliminazione delle generalizzazioni

Scelta degli identificatori primari

Si noti che in questa fase è possibile fare riferimento al costo delle operazioni precedentemente realizzato per guidare le scelte. Ad esempio, un leggero spreco di memoria legato alla non rimozione di ridondanze può essere facilmente giustificato da un guadagno in termini di prestazioni.

## Trasformazione di attributi e identificatori

Qualora siano presenti, in questa fase della progettazione, attributi ripetuti o identificatori esterni, descrivere quali trasformazioni vengono realizzate sul modello per facilitare la traduzione nello schema relazionale.

## Traduzione di entità e associazioni

Riportare in questa sezione la traduzione di entità ed associazioni nello schema relazionale.

Fornire una rappresentazione grafica del modello relazionale completo.

## Normalizzazione del modello relazionale

Effettuare la normalizzazione del modello relazionale precedentemente descritto (in forma grafica) andando a mostrare le forme 1NF, 2NF, 3NF.

# Progettazione fisica

## Utenti e privilegi

Descrivere, all’interno dell’applicazione, quali utenti sono stati previsti con quali privilegi di accesso su quali tabelle, giustificando le scelte progettuali.

## Strutture di memorizzazione

Compilare la tabella seguente indicando quali tipi di dato vengono utilizzati per memorizzare le informazioni di interesse nelle tabelle, per ciascuna tabella.

|  |  |  |
| --- | --- | --- |
| **Tabella <nome>** | | |
| **Colonna** | **Tipo di dato** | **Attributi[[2]](#footnote-1)** |
|  |  |  |

## Indici

Compilare la seguente tabella, per ciascuna tabella del database in cui sono presenti degli indici. Descrivere le motivazioni che hanno portato alla creazione di un indice.

|  |  |
| --- | --- |
| **Tabella <nome>** | |
| **Indice <nome>** | **Tipo[[3]](#footnote-2):** | |
| Colonna 1 | <nome> | |

## Trigger

Descrivere quali trigger sono stati implementati, mostrando il codice SQL per la loro instanziazione. Si faccia riferimento al fatto che il DBMS di riferimento richiede di utilizzare trigger anche per realizzare vincoli di check ed asserzioni.

## Eventi

Descrivere quali eventi sono stati implementati, mostrando il codice SQL per la loro instanziazione. Si descriva anche se gli eventi sono istanziati soltanto in fase di configurazione del sistema, o se alcuni eventi specifici vengono istanziati in maniera effimera durante l’esecuzione di alcune procedure.

## Viste

Mostrare e commentare il codice SQL necessario a creare tutte le viste necessarie per l’implementazione dell’applicazione.

## Stored Procedures e transazioni

Mostrare e commentare le stored procedure che sono state realizzate per implementare la logica applicativa delle operazioni sui dati, evidenziando quando (e perché) sono state realizzate operazioni transazionali complesse.

# Appendice: Implementazione

## Codice SQL per instanziare il database

Riportare il codice SQL necessario ad istanziare lo schema del DB. Le stored procedure, le viste, i trigger, gli eventi e tutto quello che è stato già inserito all’interno della relazione di progetto nelle sezioni precedenti non deve essere inserito in questa appendice.

Sì, avete letto bene: **riportare il codice SQL**. Frasi del tipo “il codice è nel file allegato” non rispondono alla richiesta di riportare il codice SQL.

## Codice del Front-End

Riportare (correttamente formattato) il codice C del thin client realizzato per interagire con la base di dati.

Sì, avete letto bene: **riportare il codice C**. Frasi del tipo “il codice è nei file allegati” non rispondono alla richiesta di riportare il codice C.

1. Indicare con E le entità, con R le relazioni [↑](#footnote-ref-0)
2. PK = primary key, NN = not null, UQ = unique, UN = unsigned, AI = auto increment. È ovviamente possibile specificare più di un attributo per ciascuna colonna. [↑](#footnote-ref-1)
3. IDX = index, UQ = unique, FT = full text, PR = primary. [↑](#footnote-ref-2)