



Università degli Studi dell'Insubria  
Dipartimento di Scienze Teoriche e Applicate

---

## Laboratorio Interdisciplinare B

Specifiche di Progetto  
a.a. 2022/2023

Dipartimento di Scienze Teoriche e Applicate  
[alessandra.rizzardi@uninsubria.it](mailto:alessandra.rizzardi@uninsubria.it)

---

# Laboratorio Interdisciplinare - Cos'è:

---

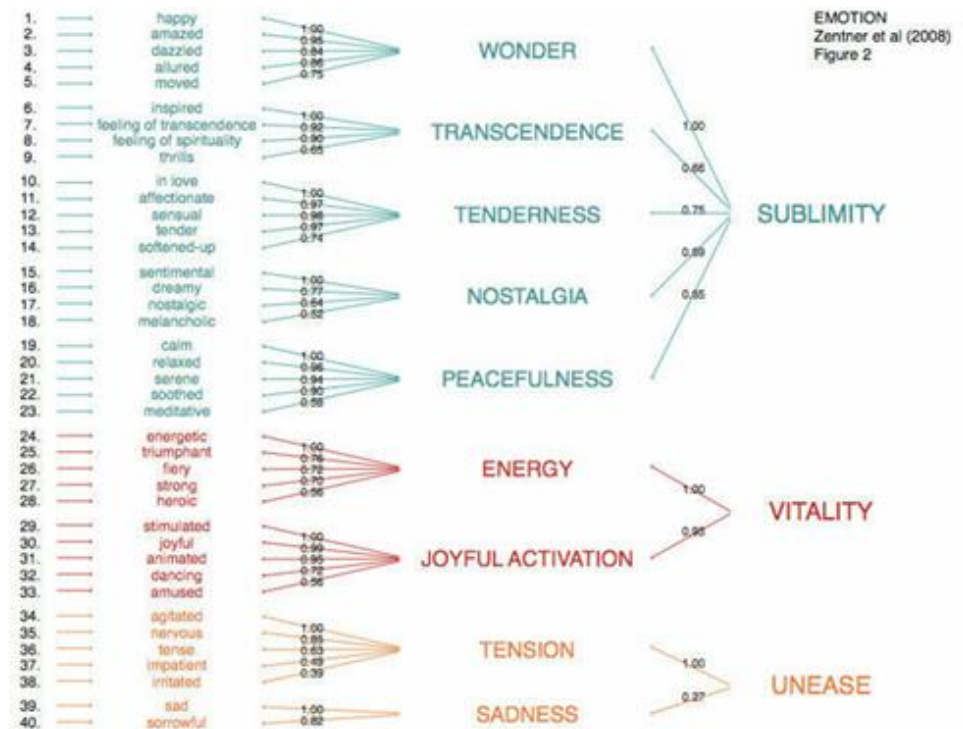
- **Obiettivi:** mettere a fattor comune e in pratica le conoscenze acquisite nei corsi di Progettazione, Sviluppo Concorrente e Distribuito, e Basi Dati
- **Come:** progettando e sviluppando un progetto software reale

# Il Progetto – Emotional songs

- ***Un sistema di annotazione di emozioni percepite durante l'ascolto di canzoni o brani musicali***
- Usiamo una scala standard per annotare le emozioni: Geneva Emotional Music Scales (GEMS)\* → scala di 9 emozioni utilizzata in letteratura

*Titolo del articolo: Emotions evoked by the sound of music: Characterization, classification and measurement.*

*Autori: Zentner, Grandjean, and Scherer*  
*Rivista: Emotion, 8, 494-521 (2008).*



\* [https://www.uibk.ac.at/psychologie/fachbereiche/pdd/personality\\_assessment/gems/about-the-gems/index.html.en](https://www.uibk.ac.at/psychologie/fachbereiche/pdd/personality_assessment/gems/about-the-gems/index.html.en)

# Il Progetto – Emotional songs

- Un sistema di annotazioni di emozioni di canzoni  
*indicano l'intensità dell'emozione percepita durante l'ascolto del brano musicale, su una scala che va da 1 (per niente) a 5 (molto)*
- Note opzionali testuali (max 256 caratteri liberi)

| Emotional category | Explanation  | Score     | Notes (max 256 characters) |
|--------------------|--|-----------|----------------------------|
| <b>Amazement</b>   | Feeling of wonder or happiness                       | 1 ..... 5 |                            |
| <b>Solemnity</b>   | Feeling of transcendence, inspiration. Thrills.      | 1 ..... 5 |                            |
| <b>Tenderness</b>  | Sensuality, affect, feeling of love                  | 1 ..... 5 |                            |
| <b>Nostalgia</b>   | Dreamy, melancholic, sentimental feelings            | 1 ..... 5 |                            |
| <b>Calmness</b>    | Relaxation, serenity, meditateness                   | 1 ..... 5 |                            |
| <b>Power</b>       | Feeling strong, heroic, triumphant, energetic        | 1 ..... 5 |                            |
| <b>Joy</b>         | Feels like dancing, bouncy feeling, animated, amused | 1 ..... 5 |                            |
| <b>Tension</b>     | Feeling Nervous, impatient, irritated                | 1 ..... 5 |                            |
| <b>Sadness</b>     | Feeling Depressed, sorrowful                         | 1 ..... 5 |                            |

## Il Progetto – Emotional songs

---

A partire da un repository di canzoni (da costruire), l'applicazione «Emotional songs» permette di:

- creare una o più playlist di brani e annotarli singolarmente con le emozioni indotte dall'ascolto, secondo una scala standard di 9 stati emozionali
- mostrare un prospetto riassuntivo sui tag emozionali degli utenti (in forma aggregata) per ciascun brano della propria playlist

# Il Progetto – Emotional songs

---

Prima di progettare l'applicazione, è necessario costruire un Database con una Tabella denominata `Canzoni`, dove memorizzare le canzoni, che deve contenere almeno i seguenti campi:

- Titolo
- Autore
- Anno

Campi opzionali:

- Album
- Durata
- Genere

## Il Progetto – Emotional songs

---

L'applicazione permette di:

1. consultare le informazioni del repository di canzoni (accesso libero)
2. registrarsi all'applicazione
3. creare playlists (solo dopo login)
4. inserire le emozioni provate nell'ascoltare ciascuna canzone della playlist  
(solo dopo login)

# Il Progetto – Emotional songs

---

1) Per consultare le informazioni di ogni canzone (non è necessario login o registrazione):

a. funzionalità di ricerca `cercaBranoMusicale()`

- ricerca per titolo (prende in input una stringa di caratteri e restituisce i brani nel cui nome compare la stringa di caratteri)
- ricerca per autore e anno (prende in input un autore e un anno e restituisce il nome del/i brano/i corrispondenti ad autore ed anno ricercato)

b. funzionalità di selezione e visualizzazione `visualizzaEmozioneBrano()`

- una volta ricercato il brano, deve essere possibile selezionare il brano e visualizzare tutte le informazioni relative al brano selezionato
- deve anche essere possibile visualizzare un prospetto riassuntivo delle emozioni associate a quel brano o l'indicazione che il brano ricercato non contiene emozioni inserite dagli utenti. Nel caso di presenza di tag emozionali per quel brano, si potranno visualizzare le emozioni associate dagli utenti, in forma aggregata, con il numero di utenti per ciascuna emozione e, ad esempio, la media del punteggio per ciascuna emozione, oltre agli eventuali commenti lasciati dagli utenti



# Il Progetto – Emotional songs

---

2) Per registrarsi all'applicazione, tramite la funzione `registrazione()`, l'utente deve inserire:

- nome e cognome
- codice fiscale
- indirizzo fisico (via/piazza, numero civico, cap, comune, provincia)
- indirizzo di posta elettronica
- userid
- password per accedere al sistema

I dati della registrazione devono essere salvati in una Tabella del DB denominata `UtentiRegistrati`

## Il Progetto – Emotional songs

---

3) Per creare la playlist, tramite la funzione `registraPlaylist()` l'utente:

- a. deve autenticarsi tramite `userid` e `password` fornite al momento della registrazione
- b. tramite la funzione `registraPlaylist()` deve inserire:
  - Nome della playlist
  - Elenco dei brani della playlist (singoli brani, brani di un autore o singoli album)

I dati di ogni playlist per ciascun utente registrato sono memorizzati in una Tabella del DB denominata `Playlist`

# Il Progetto – Emotional songs

- 4) Per inserire eventuali emozioni provate durante l'ascolto, l'utente registrato:
- a. deve autenticarsi tramite userid e password fornite al momento della registrazione
  - b. deve ricercare e selezionare la canzone ascoltata all'interno della playlist
  - c. può ora usare la funzione `inserisciEmozioniBrano()`

(la canzone e le emozioni associate dall'utente sono memorizzate nella Tabella del DB denominata `Emozioni`)

| Emotional category | Explanation  | Score     | Notes (max 256 characters) |
|--------------------|--|-----------|----------------------------|
| <b>Amazement</b>   | Feeling of wonder or happiness                       | 1 ..... 5 |                            |
| <b>Solemnity</b>   | Feeling of transcendence, inspiration. Thrills.      | 1 ..... 5 |                            |
| <b>Tenderness</b>  | Sensuality, affect, feeling of love                  | 1 ..... 5 |                            |
| <b>Nostalgia</b>   | Dreamy, melancholic, sentimental feelings            | 1 ..... 5 |                            |
| <b>Calmness</b>    | Relaxation, serenity, meditateness                   | 1 ..... 5 |                            |
| <b>Power</b>       | Feeling strong, heroic, triumphant, energetic        | 1 ..... 5 |                            |
| <b>Joy</b>         | Feels like dancing, bouncy feeling, animated, amused | 1 ..... 5 |                            |
| <b>Tension</b>     | Feeling Nervous, impatient, irritated                | 1 ..... 5 |                            |
| <b>Sadness</b>     | Feeling Depressed, sorrowful                         | 1 ..... 5 |                            |

## Il Progetto – Emotional songs

---

All'avvio l'applicazione mostra un menù iniziale dove:

Tutti possono:

- cercare brani con titolo, autore, anno
- visualizzare le emozioni associate a ciascuna canzone selezionata
- registrarsi all'applicazione

Gli utenti registrati possono:

- creare playlist ed inserirvi canzoni
- inserire le emozioni provate all'ascolto di determinate canzoni

# Il Progetto – Piattaforma ES

---

La Piattaforma Emotional Songs (ES) consiste di:

- 1.un modulo `serverES`, che interfacciandosi con un DBMS relazionale (PostgreSQL) fornisce servizi di back-end
- 2.un modulo `clientES`, che fornisce tutti i servizi e funzionalità designate per gli utilizzatori dell'applicazione.

Al lancio di `serverES` deve essere richiesto di specificare:

- 1.le credenziali per accedere al `dbES` (database di supporto all'esecuzione dei servizi della piattaforma ES)
- 2.l'host del DB

Una volta lanciato `serverES`, questo dovrà rimanere in attesa di richieste di connessione da parte di client `clientES`

L'applicazione ES dovrà essere realizzata in modo tale da supportare l'interazione in parallelo con più utenti connessi alla piattaforma da postazioni (client) differenti

# Il Progetto – Aspetti Chiave

---

- Strutture dati:
  - Canzoni
  - UtentiRegistrati
  - Playlist
  - Emozioni
- Cosa e come memorizzare su DB (e relativa progettazione)
- Interfaccia utente con interfaccia grafica
- Gestione della Concorrenza  
(i servizi dell'applicazione vengono erogati in parallelo a più utenti, e possono verificarsi accessi concorrenti a risorse condivise)

# Il Progetto – ToDo

---

1. Progettazione della Soluzione
2. Sviluppo della Soluzione Software
3. Documentazione di Progetto (2 documenti distinti)\*
  - Manuale Utente
  - Manuale Tecnico

\* Ulteriori informazioni su come scrivere la documentazione le vedremo la prossima lezione

# Il Progetto – ToDo

---

## 1. Progettazione della Soluzione

Le attività di analisi e progettazione devono essere adeguatamente documentate facendo uso del **linguaggio UML** per l'applicazione software e del modello **Entity-Relationship (ER)** per il database.

È richiesto di progettare l'applicazione avvalendosi dove possibile dell'uso di **design patterns**, e di realizzare l'applicazione con un'opportuna **interfaccia grafica**, usando il linguaggio Java e gli strumenti utili

Progettare e realizzare un database utilizzando **PostgreSQL** per la sua implementazione (<http://www.postgresql.org>) e **JDBC** per l'accesso alla base di dati da programma Java (<http://jdbc.postgresql.org/download.html>)



# Il Progetto – ToDo

---

## 2. Sviluppo della Soluzione – Database

- Si ristrutturi, se necessario, secondo le metodologie di progettazione i requisiti descritti. Si scelgano le metodologie per la costruzione dello schema ER, motivando le scelte fatte.
- Si definisca lo schema concettuale ER per il database, evidenziando le entità e le associazioni di interesse, nonché i vincoli di cardinalità e di identificazione, motivando le scelte effettuate. Altri eventuali vincoli devono essere espressi in linguaggio naturale.
- Si effettui la ristrutturazione dello schema ER motivando le scelte effettuate.
- E' richiesto di produrre un documento di analisi dei requisiti ristrutturato e documentazione associata allo schema ER (ristrutturato e non), con eventuale specifica di vincoli in linguaggio naturale.
- Si effettui la traduzione dello schema ER ristrutturato in un equivalente schema relazionale. E' richiesto di produrre la documentazione associata allo schema relazionale derivato dallo schema concettuale.
- Si realizzi il database utilizzando PostgreSQL, e SQL per la definizione dei dati, l'implementazione dei vincoli identificati, e la manipolazione dei dati, secondo le operazioni previste dall'applicazione.
- Documentare gli script SQL necessari alla creazione della base di dati e dei vincoli definiti sui dati e le query SQL a supporto dei servizi erogati da Interfacce di Programmazione.

# Il Progetto – Codice Sorgente

---

1. Il progetto deve essere sviluppato in linguaggio **Java** (versione recente, non oltre la 17) e deve essere multiplatforma
2. Il codice deve essere opportunamente commentato in formato **javadoc**
3. Il package `emotionalsongs` deve essere definito e deve contenere le relative classi (ulteriori package sono ammessi)
4. Il `main` per l'esecuzione dell'applicazione deve essere contenuto nella classe di nome `EmotionalSongs` del package `emotionalsongs`
5. L'intestazione di tutti i file `*.java` devono contenere *nome, cognome, num. matricola, sede (VA o CO)* degli autori del progetto

# Il Progetto – ToDo

---

## 3. Documentazione di Progetto

- (Seguirà lezione ad hoc su come scrivere un corretto documento utente e manuale tecnico)

# Il Progetto – Consegna

---

Il progetto deve essere consegnato **come link a una directory OneDrive dell'Università o Google Drive del project manager** e al link si dovrà trovare **una dir compressa denominata con *cognome\_matricola* del project manager del team**. La cartella compressa dovrà contenere:

- un file dal nome **autori.txt** contenente cognome, nome, numero di matricola e sede (VA o CO) di ogni membro del team
- la directory **doc** contenente il manuale utente, il manuale tecnico in formato .pdf, e tutti gli artefatti (diagrammi ER, UML) prodotti
- la directory **src** contenente il codice sorgente del progetto
- file di build Apache Ant (<http://ant.apache.org/>) o Maven (<https://maven.apache.org/>) per compilare il progetto, lanciare il server e i client, creare il database, creare la documentazione javadoc, etc.
- eventuali librerie necessarie alla compilazione e/o all'esecuzione
- file README con indicazioni precise sull'installazione e sulla compilazione, specificando i comandi Ant/Maven da utilizzare, ed indicazioni di particolari librerie, usate in modo non standard.

## Il Progetto – Consegna

---

3. La consegna viene fatta tramite invio link all'indirizzo [alessandra.rizzardi@uninsubria.it](mailto:alessandra.rizzardi@uninsubria.it) con oggetto email: **Progetto Lab B**
4. Responsabile della consegna sarà il project manager tramite email istituzionale di ateneo
5. Le date di consegna verranno comunicate di volta in volta (indicativamente una decina di giorni prima della data di appello d'esame)

# Il Progetto – Dubbi e domande

---

Per ogni dubbio sulle specifiche che non sia chiarito durante la lezione di presentazione del progetto, è possibile scrivere sul forum della pagina del corso e le varie domande saranno indirizzate durante la prossima attività laboratoriale.

Il **forum** si chiama "**Specifiche progetto lab B**"

## Il Progetto – Valutazione

---

- In fase di discussione orale verrà:
  - richiesto allo studente di saper argomentare in modo opportuno le scelte progettuali, algoritmiche, e implementative adottate
  - verificata l'effettiva padronanza delle tecniche utilizzate attraverso una serie di domande
- La valutazione terrà conto dei seguenti fattori:
  - l'aderenza del sistema realizzato ai requisiti proposti
  - la qualità dei documenti di analisi e progettazione prodotti sia per la realizzazione del software che per il database (correttezza sintattica, semantica, completezza e leggibilità, minimalità dello schema logico)
  - le scelte algoritmiche e di progettazione effettuate (design pattern)
  - la qualità del codice sorgente prodotto (funzionalità, correttezza, facilità d'uso).