

Scelte Progettuali per il sistema di gestione ToDo

Obbiettivo del Progetto

Il sistema nasce con l'obiettivo di **organizzare e gestire attività personali (ToDo)**. Deve permettere a uno o più utenti di:
Autenticarsi nel sistema, Organizzare attività in bacheche, Condividere ToDo con altri utenti, Gestire scadenze, descrizioni, immagini e stato dei ToDo e Ricercare e filtrare i ToDo.

Classi Identificate

Ho scelto di creare una *classe* **Utente** per rappresentare l'utente registrato con le proprie credenziali (username e password).

Successivamente ho creato la *classe* **Bacheca** per rappresentare il contenitore di ToDo (attività), dove ogni utente ha tre bacheche fisse (Università, Lavoro e TempoLibero) modellate come *enumerazione* **TitoloBacheca**.

Per rappresentare le attività ho creato la classe **ToDo** (classe centrale del sistema) che contiene informazioni come titolo, descrizione, scadenza, colore, URL, immagine, l'autore, e infine lo stato, che ho modellato anch'essi come *enumerazione* **StatoToDo** che indica se l'attività è stata Completata oppure Non Completata.

Infine ho creato la classe **VistaToDo** in modo da permettere la condivisione di ToDo tra più utenti, la visione personalizzata (bacheca e posizione nella bacheca).

Quindi, contiene informazioni personalizzate per ciascun utente, come:

in quale bacheca l'utente vede il ToDo

in che posizione si trova nella bacheca

Ad esempio se un utente sposta il ToDo da “Università” a “Lavoro”:

Il sistema modifica `VistaToDo.bacheca` corrispondente a quell'utente (non quella degli altri).

Quindi ogni utente può decidere in quale delle sue 3 bacheche mettere un
ToDo condiviso. Il ToDo è lo stesso, ma le viste possono essere
indipendenti nei contenuti di bacheca e posizione

Ho utilizzato le **enumerazioni** per mantenere valori fissi e condivisi:

TitoloBacheca: Università, Lavoro, Tempo Libero

StatoToDo: Completato, Non Completato

Gli attributi delle enumerazioni li ho impostati come **public** perché devono
essere visibili ovunque servano, e come **classifier** perché ogni valore è
definito una sola volta e viene riutilizzato ovunque.