

Il nostro applicativo si presenta con 6 package :

- MODEL
- DAO
- DATABASE
- GUI
- CONTROLLER
- POSTGRESDAO

Rispettivamente nel model troviamo le nostre classi principali, con i metodi set e get per ogni attributo. In dao abbiamo l'interfaccia ListinoDao che viene implementata dalla classe ListinoPostgresDao che si trova nel package postgresdao. Poi troviamo il controller che fa da "passante" tra il model e la gui (nel package controller troviamo anche le exception), in quest'ultima troviamo tutti i form con le rispettive classi in java. Infine abbiamo il package database, dove abbiamo la classe ConnessioneDatabase in cui c'è l'avvenuta connessione al database e il file.txt dove ci sono le effettive credenziali per la avvenuta connessione ad esso. In allegato possiamo trovare i 2 UML richiesti :

- UML DEL DOMINIO DEL PROBLEMA
- UML DEL DOMINIO DELLA SOLUZIONE

Dove nel primo possiamo trovare tutte le nostre classi, senza attributi ma solo con responsabilità e ruoli, mentre nel secondo troviamo tutte le classi (con i metodi) da noi implementate compresi i package riportati sopra.

Come possiamo vedere nel uml del dominio del problema non compare la classe autore e quella testo, di seguito le meritate spiegazioni :

- Per quanto riguarda la classe autore abbiamo preferito eliminarla poiché ci sembrava una ridondanza, inoltre avendo solo utenti abbiamo reso più realistico il nostro mini-mondo, in modo da non avere sempre e solo o autori o utenti, ma utenti che se non creano pagine o non ne hanno mai creato una rimangono tali, mentre in caso contrario possono essere considerati autori (un'ultima motivazione per questa scelta è stata anche quella per avere una maggiore efficienza di comunicazione con il database).
- Invece per la classe testo abbiamo preferito eliminarla per gli stessi motivi di autore, ci sembrava ridondante poiché, potevamo anche semplicemente dire che una pagina è composta da frasi e il testo è "accorpato" in pagina (un'ultima motivazione per questa scelta è stata anche quella per avere una maggiore efficienza di comunicazione con il database).