

---

# L-BOOK

---

Progetto di Programmazione ad Oggetti

ANNO 2016/2017  
MIGLIORIN FRANCESCO ANTONIO  
MATRICOLA 1096825

## Indice

1. Abstract.....	3
2. Rispetto dei requisiti.....	3
3. Logica del programma.....	4
4. Grafica del programma.....	5
5. Stream di Input/Output.....	5
6. Polimorfismo.....	6
7. Contenitore.....	6
8. Dati di esempio.....	7
9. Terze Parti.....	7
10. Ambiente di Sviluppo.....	7
11. Prima Installazione.....	8
12. Manuale Utente.....	8

## 1. Abstract

L-Book nasce come un programma gestionale pensato per l'amministrazione di una libreria scolastica. È creato in modo tale che possa essere utilizzato dagli studenti, dagli insegnanti e dal bibliotecario, inoltre essendo la realtà scolastica molto attiva negli scambi internazionali, è stato deciso di supportare il multilingua. Il programma è stato pensato per essere utilizzato principalmente dagli studenti che potranno controllare i propri prestiti, lo stato e le informazioni dei libri. I professori avranno le medesime funzioni con la possibilità aggiuntiva di poter controllare i prestiti dei propri studenti. Il programma sarà amministrato dalla figura del bibliotecario, l'unico con i permessi di scrittura, che gestirà i libri, gli utenti e si occuperà dell'inserimento dei prestiti e del loro riscatto. In molte scuole, soprattutto in quelle più piccole, la figura del bibliotecario combacia con quella di un professore. Questa situazione non è ricreabile in L-Book dove il bibliotecario e i professori sono due figure distinte, però grazie alle azioni messe a sua disposizione l'amministratore può comunque svolgere le stesse attività dell'insegnante. Per queste due figure può essere utile avere i dati da elaborare con programmi esterni, per questo motivo è possibile esportarli in un file di lettura semplice come il CSV. L-Book è stato creato in modo tale da poter essere usato su più computer utilizzando lo stesso database, presente in una cartella condivisa. La creazione di quest'ultima è affidata ad una terza parte, al di fuori del programma, in modo tale da lasciargli la libertà di configurare la condivisione ed è un'operazione possibile dai vari sistemi operativi Windows, Linux e Macintosh.

## 2. Rispetto dei requisiti

Nella specifica del progetto sono presenti una serie di vincoli obbligatori che saranno qui sotto brevemente discussi:

- 1) Dal pannello di amministrazione è possibile accedere a tutti i dati salvati su file e a sua volta scrivere i nuovi dati immessi o modificati. Infatti il pannello offre la possibilità di creazione / modifica e rimozione dei dati sulla memoria, le modifiche verranno salvate su file solo dopo aver cliccato l'apposito bottone nel pannello di amministrazione. Si ricorda che tali opzioni si possono effettuare anche sugli utenti e che la registrazione è possibile solo dal pannello di amministrazione come richiesto dal Punto C. Dalla pagina di registrazione l'Amministratore può registrare professori e studenti, solo al primo avvio sarà possibile registrare un Amministratore se questo non è presente. Inoltre il contenitore scelto è di tipo Template di conseguenza può memorizzare tipi di diversa natura.
- 2) Ogni utente accede al contenitore in base ai suoi permessi: tutti gli utenti possono accedere alla lista dei libri in sola lettura e ai dati relativi al proprio account (solo nel caso gli siano state create le credenziali di login). Per quanto riguarda i prestiti gli studenti possono visionare solo i loro mentre i professori accedono anche a quelli dei loro studenti. L'amministratore ha accesso a ogni dato ed è l'unico con permessi di scrittura.
- 3) La grafica di L-Book è stata interamente realizzata utilizzando QtCreator e il designer che mette a disposizione.

Un ulteriore requisito, che non è presente nella specifica del progetto, ma detto a voce dal professore durante la presentazione del progetto è: "È possibile utilizzare solo librerie Qt."

Questo ulteriore vincolo è stato rispettato realizzando l'intero progetto solo utilizzando le librerie che Qt mette a disposizione e quindi escludendo tutte le altre, compresa la Libreria Standard C++. Per concludere ogni riferimento a Qt è inteso per la versione 5.5, se non diversamente specificato, rispettando la versione indicata nella specifica del progetto.

### 3. Logica del programma

La "logica del programma" è quella parte di L-Book destinata alla gestione ed elaborazione dei dati che vengono richiesti dai vari utenti.

Il codice si trova nella sottocartella ../Lib/ della cartella del progetto ed è strutturata in 13 classi.

#### Descrizione delle classi:

- **DataCollector:** questa classe contiene le liste di tutti gli utenti, dei libri e dei prestiti che hanno una correlazione logica tra loro.
- **Library/Loans/Users:** queste tre classi contengono rispettivamente la lista delle pubblicazioni, dei prestiti e degli utenti. Consentendo di interagire con questi dati tramite una serie di funzioni.
- **Loan:** è una classe base che contiene tutte le informazioni relative ad un prestito.
- **Account:** questa classe raggruppa tutte le classi che contengono le informazioni di un utente e sono suddivise in:
  - **account\_permission:** che contiene i permessi dell'utente
  - **userinfo:** che contiene le generalità dell'utente
  - **username:** che contiene le credenziali d'accesso dell'utente, dove la password è memorizzata già criptata secondo la funzione hash MD5.
- **Publication:** è una classe astratta che contiene le generalità in comune tra libri e periodici, che viene poi estesa dalle due sottoclassi:
  - **book:** la classe derivata che contiene tutte le informazioni relative ad un libro
  - **periodic:** la classe derivata che contiene tutte le informazioni relative ad un periodico.

In più contiene la classe:

- **publication\_permission:** che contiene i permessi di una pubblicazione.

## 4. Grafica del programma

La grafica di L-Bokk è stata realizzata utilizzando il tool QtDesigner creando Widget che si adattano in base ai permessi dell'utente. Il codice si trova nella sottocartella ../Gui/ della cartella del progetto ed è strutturata in 12 classi.

### Descrizione delle classi:

- **MainWindow:** è la finestra principale del programma dove è possibile accedere all'elenco delle pubblicazioni, cambiare lingua, aggiornare i dati ed effettuare il login per entrare nell'area utente.
- **AdminMainWindow:** è la finestra principale dell'area amministrazione dove è possibile amministrare le pubblicazioni, gli utenti e i prestiti. L'amministratore accede automaticamente dopo aver effettuato il login.
- **Setup:** finestra d'installazione del programma che appare solo durante il primo utilizzo del programma per raccogliere le informazioni base.
- **PublicationList/UserList/LoanList:** sono delle finestre di ricerca dov'è possibile filtrare i contenuti con delle stringhe chiave.
- **ShowPublication/ShowUser/ShowLoan:** sono delle finestre dove vengono visualizzate le informazioni relative alle rispettive pubblicazioni, utenti e prestiti scelti.
- **ModifyPublication/ModifyUser:** sono delle finestre dov'è possibile modificare l'oggetto in questione.
- **MakeLoan:** da questa finestra è possibile creare un prestito.

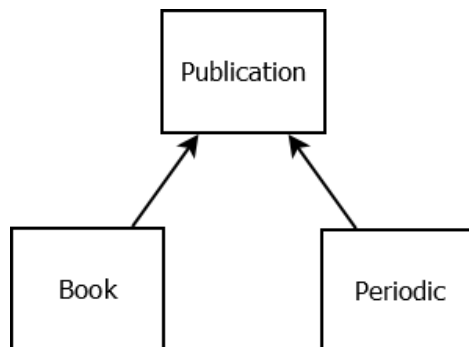
## 5. Stream di Input/Output

La gestione dell'I/O è affidata alla classe IOStream presente nella cartella ../Stream/ che si occupa dell'input di file di tipo .Json per i dati e .qm per gestire la lingua del programma, mentre fornisce in output file .Json e se l'utente abilitato lo richiede crea un file .csv per un'elaborazione esterna.

La gestione dei dati del programma è stata realizzata in Json, che è stato scelto per la sua semplicità sia di lettura per le persone che di scrittura e analisi per una macchina. L'unica difetto riscontrato è stato il fatto che Json non supporta il polimorfismo e per questa ragione è stato necessario aggiungere un attributo aggiuntivo per poter distinguere la struttura polimorfa.

## 6. Polimorfismo

La gerarchia polimorfa utilizzata nel programma è la seguente:



Al vertice troviamo la classe astratta *Publication* che viene estesa dalle classi *Book* e *Periodic*. Entrambi le sottoclassi implementano il metodo virtuale puro *setISBN\_13* ( `const QString &` ) gestendo il parametro, della superclasse, per la memorizzazione dell' ISBN che per questo motivo è stato marcato `protected`. La differenza principale sta nell'input che nel caso dei periodici può essere anche il codice ISSN. Viene gestita in modo diverso anche la funzione virtuale pura *showIdentification()* che fornisce un codice univoco che identifica la pubblicazione. Nel caso dei libri è il semplice codice ISBN 13 mentre nelle riviste viene aggiunto anche il codice identificativo composta da cinque cifre. Nel contenitore viene inserito il puntatore alla loro superclasse e l'eventuale distinzione nella sottoclasse viene fatta solo in alcune funzioni di ricerca specifiche o nel caso sia necessario mostrarne i campi dati nella Gui.

## 7. Contenitore

Disponendo della libera scelta se creare o meno il contenitore è stato scelto di utilizzarne uno già esistente. Infatti un contenitore già esistente risulta senz'altro ottimizzato per la velocità, la gestione della memoria e un'espansione minore del codice eseguibile. Un fattore essenziale per un programma come L-Book che in un utilizzo reale deve contenere una notevole mole di dati e deve essere in grado di fare ricerche rapide. La scelta è ricaduta sui 3 principali container messi a disposizione da Qt: `QList<T>`, `QLinkedList<T>`, and `QVector<T>`. Alla fine è stata scelta la classe `QList<T>` in quanto, nella maggior parte dei casi, risulta più conveniente della classe `QLinkedList` e più veloce del `QVector`. Tuttavia è stata presa in considerazione anche l'attuale documentazione, la 5.9, dove viene chiaramente detto che `QVector<T>` dovrebbe essere la scelta di default perché normalmente fornisce prestazioni migliori. Tenendo in considerazione questo fatto la classe contenitore è stata incapsulata dentro le classi *Library*, *Loans* e *Users* in questo modo risulterà sufficiente cambiare solo queste classi nel caso si voglia passare da `QList<T>` a `QVector<T>` nel caso si voglia ottenere prestazioni migliori compilando il sorgente con la versione 5.9 di Qt.

## 8.Dati di esempio

Nella cartella ../Data/ è presente un piccolo database che permette la prova del programma. Tutti i libri e i periodici sono reali, mentre gli utenti sono immaginari e ogni somiglianza con persone reali, viventi o defunte, è puramente casuale. Nella sotto cartella ../Data/languages/ troviamo i file di lingua per poter provare la funzionalità multilingua di L-Book. I dati di login per gli utenti sono:

Nome Utente	Password
admin	admin
professor	professor
studentX *	studentX *

\*Dove X indica il numero dello studente che va da 1 a 3

## 9.Terze parti

La scritta L-Book in tutto il programma, e le iniziali L-B del logo, sono state realizzate utilizzando il font "A Charming Font". Non potendo avere la certezza che tale font sia installato in ogni computer che desideri provare il sorgente, il font viene fornito al compilatore direttamente dal file .ttf presente nella cartella ../Res/. Tale file è stato scaricato gratuitamente al seguente indirizzo:

<http://www.dafont.com/it/a-charming-font.font>

## 10.Ambiente di Sviluppo

- Sistema Operativo: Windows 7 Professional N 64-bit
  - Compilatore: MinGW 4.8.1 (32-bit)
  - Versione Qt: 5.5.1
- 
- Sistema Operativo: Ubuntu 16.04.04 LTS
  - Compilatore: GCC 5.4.0
  - Versione Qt: 5.5.1

Il programma è stato anche compilato anche nel computer di laboratorio.

## **11. Prima Installazione**

Al primo avvio il programma ci presenterà la finestra di setup dove ci chiederà di scegliere se gestire i file localmente (opzione pensata per chi vuole utilizzare un solo computer), oppure in modo condiviso e quindi ci chiederà di selezionare una cartella dove leggerà e scriverà i dati. Se la cartella contiene già dei dati generati da un altro programma L-Book utilizzato su un altro computer o quelli di esempio il programma provvederà autonomamente a caricarli. Altrimenti se non ci sono file, o manca l'amministratore, il programma procederà chiedendoci di registrarne uno. Finita questa procedura si aprirà la finestra iniziale.

Il programma come lingua di default è in inglese se si vuole cambiare la lingua è necessario fornirgli la cartella ../Data/languages/ nella stessa directory dov'è presente l'eseguibile oppure nella directory dei dati condivisi( in questo caso la finestra di setup sarà in inglese ) .

## **12. Manuale Utente**

La Gui è stata creata in modo da risultare semplice e user friendly. Ogni bottone descrive l'azione che andrà ad eseguire, chiedendo eventualmente prima conferma, e spiegando meglio la sua funzione. L'unica azione un po' celata è la possibilità di ottenere informazioni aggiuntive eseguendo un doppio click su un determinato elemento di una tabella. Per evidenziare questa possibilità il cursore del mouse diventa una "manina" come quella dei link ipertestuali. In diverse finestre grafiche è presente un label con delle note, che servono a indicarci i campi richiesti o i campi che possiamo immettere per una ricerca.