1.1.1 Componenti off-the-shelf

Al fine di evitare che i tempi di sviluppo del software vengano allungati più del necessario, si è deciso di utilizzare diversi software già pronti.

Per la gestione persistente dei dati si è deciso di ricorrere a MySQL come DBMS poiché gli sviluppatori hanno già esperienza nell’utilizzo di questo software.

Per scambiare informazioni ed eseguire azioni tra il database e il software si utilizzerà JDBC che è un componente il cui utilizzo ed efficienza sul mercato è ampiamente documentato.

Per la creazione delle interfacce verrà utilizzato il GUIDesigner basato su Swing e awt.

Tutte le componenti selezionate sono gratuite ed open source, scelta in linea con i requisiti di costo.

1.1.2 Design patterns

Per velocizzare lo sviluppo del sistema abbiamo utilizzato alcuni design patterns che risolvono problemi tipici anche del nostro progetto.

La scelta è ricaduta su tre Design Pattern, specificamente: Composite, Bridge e Observer.

Il design Composite per definizione è un pattern basato sulla gestione di oggetti composti come se fossero un solo oggetto semplice. Infatti, è stato scelto poiché si adatta perfettamente alle necessità di definizione delle interfacce e della gestione delle componenti di cui sono composte.

Poiché nel nostro software verranno utilizzati bottoni ed eventi, si utilizzerà il pattern Observer il quale ci permetterà di avere del codice che alla ricezione di un evento cambi lo stato di tutti gli oggetti interessati e invochi i metodi necessari al proseguo delle attività.

Per ultimo, verrà utilizzato il .