**DAL Package**

**BaseDAL**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAL;

import java.sql.CallableStatement;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

/\*\*

\*

\* @author massami

\*/

public class BaseDAL {

private static final String DATABASE\_NAME = "jdbc:mysql://localhost:3306/librarysystem\_db?useSSL=false, root, root";

static Connection con;

/\*\*

\* Get the configuration for the data base connection

\* @return the connection to the database

\* @throws Exception

\*/

public static Connection getConnection() throws Exception {

try {

if (con == null) {

Class.forName("com.mysql.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/librarysystem\_db", "root", "root");

}

return con;

} catch (SQLException ex) {

throw new Exception(ex.getMessage());

}

}

/\*\*

\* Prepare the call to the database

\* @param sql Query

\* @return statement

\* @throws Exception

\*/

public static CallableStatement getStatement(String sql) throws Exception{

Connection con = getConnection();

return con.prepareCall(sql);

}

}

**BookDAL**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAL;

import Model.Book;

import Model.Catalogue;

import java.sql.CallableStatement;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

/\*\*

\*

\* @author massami

\*/

public class BookDAL extends BaseDAL {

/\*\*

\* Insert new book to the database

\* @param book new object to be inserted

\* @throws Exception

\*/

public static void addBook(Book book) throws Exception {

String sql = "call sp\_addBook(?,?,?,?,?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setString(1, book.getAuthor());

st.setString(2, book.getGenre());

st.setString(3, book.getIsbn());

st.setString(4, book.getTitle());

st.setString(5, book.getDescription());

st.setString(6, book.getReleaseDate());

st.setString(7, book.getStatus());

st.setString(8, book.getNote());

st.setString(9, book.getLanguage());

st.executeQuery();

}

/\*\*

\* Update the information of the book

\* @param book object to be updated

\* @throws Exception

\*/

public static void updateBook(Book book) throws Exception {

String sql = "call sp\_updateBook(?,?,?,?,?,?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setInt(1, book.getId());

st.setString(2, book.getAuthor());

st.setString(3, book.getGenre());

st.setString(4, book.getIsbn());

st.setString(5, book.getTitle());

st.setString(6, book.getDescription());

st.setString(7, book.getReleaseDate());

st.setString(8, book.getStatus());

st.setString(9, book.getNote());

st.setString(10, book.getLanguage());

st.executeQuery();

}

/\*\*

\* Get all the books

\* @return all the books

\* @throws Exception

\*/

public static ArrayList<Catalogue> getAllBook() throws Exception {

ArrayList<Catalogue> catalogueList = new ArrayList();

String sql = "call sp\_searchBook()";

ResultSet rs = getStatement(sql).executeQuery();

while (rs.next()) {

Book book = new Book(rs.getInt("catalogue\_id"), rs.getInt("book\_id"), rs.getString("author"), rs.getString("genre"), rs.getString("isbn"), rs.getString("title"), rs.getString("description"), rs.getString("release\_date"), rs.getString("status"), rs.getString("note"), rs.getString("language"));

catalogueList.add(book);

}

return catalogueList;

}

/\*\*

\* Get the book filtered by the title

\* @param title property to be searched

\* @return list of books with the selected title

\* @throws Exception

\*/

public static ArrayList<Catalogue> getBookByTitle(String title) throws Exception {

ArrayList<Catalogue> catalogueList = new ArrayList();

String sql = "call sp\_searchBookByTitle(?)";

CallableStatement st = getStatement(sql);

st.setString(1, title);

ResultSet rs = st.executeQuery();

while (rs.next()) {

Book book = new Book(rs.getInt("catalogue\_id"), rs.getInt("book\_id"), rs.getString("author"), rs.getString("genre"), rs.getString("isbn"), rs.getString("title"), rs.getString("description"), rs.getString("release\_date"), rs.getString("status"), rs.getString("note"), rs.getString("language"));

catalogueList.add(book);

}

return catalogueList;

}

/\*\*

\* Get only the available books for borrowing

\* @return list of available books

\* @throws Exception

\*/

public static ArrayList<Catalogue> getAvailableBook() throws Exception{

ArrayList<Catalogue> catalogueList = new ArrayList();

String sql = "call sp\_searchAvailableBook()";

ResultSet rs = getStatement(sql).executeQuery();

while (rs.next()) {

Book book = new Book(rs.getInt("catalogue\_id"), rs.getInt("book\_id"), rs.getString("author"), rs.getString("genre"), rs.getString("isbn"), rs.getString("title"), rs.getString("description"), rs.getString("release\_date"), rs.getString("status"), rs.getString("note"), rs.getString("language"));

catalogueList.add(book);

}

return catalogueList;

}

/\*\*

\* Get available books by the filtered title

\* @param title filter

\* @return

\* @throws Exception

\*/

public static ArrayList<Catalogue> getAvailableBookByTitle(String title) throws Exception{

ArrayList<Catalogue> catalogueList = new ArrayList();

String sql = "call sp\_searchAvailableBookByTitle(?)";

CallableStatement st = getStatement(sql);

st.setString(1, title);

ResultSet rs = st.executeQuery();

while (rs.next()) {

Book book = new Book(rs.getInt("catalogue\_id"), rs.getInt("book\_id"), rs.getString("author"), rs.getString("genre"), rs.getString("isbn"), rs.getString("title"), rs.getString("description"), rs.getString("release\_date"), rs.getString("status"), rs.getString("note"), rs.getString("language"));

catalogueList.add(book);

}

return catalogueList;

}

/\*\*

\* Book logical delete

\* @param id id of the book

\* @throws Exception

\*/

public static void deleteBook(int id) throws Exception {

String sql = "call sp\_deleteBook(?)";

CallableStatement st = getStatement(sql);

st.setInt(1, id);

st.executeQuery();

}

}

**BorrowDAL**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAL;

import static DAL.BaseDAL.getStatement;

import Model.Book;

import Model.Borrow;

import Model.Catalogue;

import java.sql.CallableStatement;

import java.sql.ResultSet;

import java.util.ArrayList;

/\*\*

\*

\* @author massami

\*/

public class BorrowDAL extends BaseDAL{

/\*\*

\* Save data of the borrowed transaction

\* @param borrow information about the borrow

\* @throws Exception

\*/

public static void addBorrow(Borrow borrow) throws Exception{

String sql = "call sp\_addBorrow(?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setString(1, borrow.getBorrowDate());

st.setString(2, borrow.getDueDate());

st.setString(3, borrow.getReturnDate());

st.setInt(4, borrow.getMemberId());

st.setInt(5, borrow.getCatalogueId());

st.executeQuery();

}

/\*\*

\* Save data of the returned item

\* @param borrow data of the returned item

\* @throws Exception

\*/

public static void returnItem(Borrow borrow ) throws Exception{

String sql = "call sp\_returnItem(?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setString(1, borrow.getBorrowDate());

st.setString(2, borrow.getDueDate());

st.setString(3, borrow.getReturnDate());

st.setInt(4, borrow.getMemberId());

st.setInt(5, borrow.getCatalogueId());

st.executeQuery();

}

/\*\*

\* Get the ID of the selected member

\* @param name of the member

\* @return id

\* @throws Exception

\*/

public static int getMemberId(String name) throws Exception {

String sql = "call sp\_searchMemberId(?)";

CallableStatement st = getStatement(sql);

st.setString(1, name);

ResultSet rs = st.executeQuery();

if(rs.next())

return rs.getInt("member\_id");

else

return 0;

}

/\*\*

\* Get the items that were borrow by the members

\* @param name

\* @return list of borrowed items

\* @throws Exception

\*/

public static ArrayList<Catalogue> getBorrowedItem(String name) throws Exception{

ArrayList<Catalogue> catalogueList = new ArrayList<Catalogue>();

String sql = "call sp\_searchBorrowedItem(?)";

CallableStatement st = getStatement(sql);

st.setString(1, name);

ResultSet rs = st.executeQuery();

while(rs.next()){

Catalogue catalogue = new Book();

catalogue.setId(rs.getInt("catalogue\_id"));

catalogue.setTitle(rs.getString("title"));

catalogue.setDescription( rs.getString("description"));

catalogue.setReleaseDate(rs.getString("release\_date"));

catalogue.setStatus(rs.getString("status"));

catalogue.setNote(rs.getString("note"));

catalogue.setLanguage(rs.getString("language"));

catalogueList.add(catalogue);

}

return catalogueList;

}

}

**DvdDAL**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAL;

import Model.Catalogue;

import Model.Dvd;

import java.sql.CallableStatement;

import java.sql.Connection;

import java.sql.ResultSet;

import java.util.ArrayList;

/\*\*

\*

\* @author massami

\*/

public class DvdDAL extends BaseDAL {

/\*\*

\* Insert DVD information to the database

\* @param dvd

\* @throws Exception

\*/

public static void addDvd(Dvd dvd) throws Exception {

String sql = "call sp\_addDvd(?,?,?,?,?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setString(1, dvd.getDistributor());

st.setString(2, dvd.getCast());

st.setString(3, dvd.getCredits());

st.setString(4, dvd.getTitle());

st.setString(5, dvd.getDescription());

st.setString(6, dvd.getReleaseDate());

st.setString(7, dvd.getStatus());

st.setString(8, dvd.getNote());

st.setString(9, dvd.getLanguage());

st.executeQuery();

}

/\*\*

\* Update the information of the DVD to the database

\* @param dvd

\* @throws Exception

\*/

public static void updateDvd(Dvd dvd) throws Exception {

String sql = "call sp\_updateDvd(?,?,?,?,?,?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setInt(1, dvd.getId());

st.setString(2, dvd.getDistributor());

st.setString(3, dvd.getCast());

st.setString(4, dvd.getCredits());

st.setString(5, dvd.getTitle());

st.setString(6, dvd.getDescription());

st.setString(7, dvd.getReleaseDate());

st.setString(8, dvd.getStatus());

st.setString(9, dvd.getNote());

st.setString(10, dvd.getLanguage());

st.executeQuery();

}

/\*\*

\* Get the list of all DVDs

\* @return list of DVDs

\* @throws Exception

\*/

public static ArrayList<Catalogue> getAllDvd() throws Exception {

ArrayList<Catalogue> catalogueList = new ArrayList();

String sql = "call sp\_searchDvd()";

ResultSet rs = getStatement(sql).executeQuery();

while (rs.next()) {

Dvd dvd = new Dvd(rs.getInt("catalogue\_id") ,rs.getInt("dvd\_id"), rs.getString("distributor"), rs.getString("cast"), rs.getString("credits"), rs.getString("title"), rs.getString("description"), rs.getString("release\_date"), rs.getString("status"), rs.getString("note"), rs.getString("language"));

catalogueList.add(dvd);

}

return catalogueList;

}

/\*\*

\* Get the list of DVDs filtered by the title

\* @param title filter

\* @return list of DVDs

\* @throws Exception

\*/

public static ArrayList<Catalogue> getDvdByTitle(String title) throws Exception {

ArrayList<Catalogue> catalogueList = new ArrayList();

String sql = "call sp\_searchDvdByTitle(?)";

CallableStatement st = getStatement(sql);

st.setString(1, title);

ResultSet rs = st.executeQuery();

while (rs.next()) {

Dvd dvd = new Dvd(rs.getInt("catalogue\_id"), rs.getInt("dvd\_id"), rs.getString("distributor"), rs.getString("cast"), rs.getString("credits"), rs.getString("title"), rs.getString("description"), rs.getString("release\_date"), rs.getString("status"), rs.getString("note"), rs.getString("language"));

catalogueList.add(dvd);

}

return catalogueList;

}

/\*\*

\* Get the list of available DVDs for borrowing

\* @return list of DVDs

\* @throws Exception

\*/

public static ArrayList<Catalogue> getAvailableDvd() throws Exception {

ArrayList<Catalogue> catalogueList = new ArrayList();

String sql = "call sp\_searchAvailableDvd()";

ResultSet rs = getStatement(sql).executeQuery();

while (rs.next()) {

Dvd dvd = new Dvd(rs.getInt("catalogue\_id") ,rs.getInt("dvd\_id"), rs.getString("distributor"), rs.getString("cast"), rs.getString("credits"), rs.getString("title"), rs.getString("description"), rs.getString("release\_date"), rs.getString("status"), rs.getString("note"), rs.getString("language"));

catalogueList.add(dvd);

}

return catalogueList;

}

/\*\*

\* Get the list of available DVDs filtered by title

\* @param title filter

\* @return list of DVDs

\* @throws Exception

\*/

public static ArrayList<Catalogue> getAvailableDvdByTitle(String title) throws Exception {

ArrayList<Catalogue> catalogueList = new ArrayList();

String sql = "call sp\_searchAvailableDvdByTitle(?)";

CallableStatement st = getStatement(sql);

st.setString(1, title);

ResultSet rs = st.executeQuery();

while (rs.next()) {

Dvd dvd = new Dvd(rs.getInt("catalogue\_id"), rs.getInt("dvd\_id"), rs.getString("distributor"), rs.getString("cast"), rs.getString("credits"), rs.getString("title"), rs.getString("description"), rs.getString("release\_date"), rs.getString("status"), rs.getString("note"), rs.getString("language"));

catalogueList.add(dvd);

}

return catalogueList;

}

/\*\*

\* Local delete of the DVD object

\* @param id

\* @throws Exception

\*/

public static void deleteDvd(int id) throws Exception {

String sql = "call sp\_deleteDvd(?)";

CallableStatement st = getStatement(sql);

st.setInt(1, id);

st.executeQuery();

}

}

**HistoryDAL**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAL;

import static DAL.BaseDAL.getStatement;

import Model.History;

import java.sql.CallableStatement;

import java.sql.ResultSet;

import java.util.ArrayList;

/\*\*

\*

\* @author massami

\*/

public class HistoryDAL extends BaseDAL{

/\*\*

\* Get the transaction history of the selected member

\* @param name

\* @return List of borrowed items

\* @throws Exception

\*/

public static ArrayList<History> getTransactionHistory(String name) throws Exception{

ArrayList<History> historyList = new ArrayList();

String sql = "call sp\_searchTransactionHistory(?)";

CallableStatement st = getStatement(sql);

st.setString(1, name);

ResultSet rs = st.executeQuery();

while (rs.next()) {

History history = new History(rs.getString("title"), rs.getString("due\_date"), rs.getString("borrow\_date"), rs.getString("name"));

historyList.add(history);

}

return historyList;

}

}

**MemberDAL**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAL;

import Model.Member;

import Model.Person;

import java.sql.CallableStatement;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.Hashtable;

import java.util.logging.Level;

import java.util.logging.Logger;

/\*\*

\*

\* @author Rom

\*/

public class MemberDAL extends BaseDAL {

/\*\*

\* Insert the member to the database

\* @param member selected member

\* @throws Exception

\*/

public static void addMember(Member member) throws Exception {

String sql = "call sp\_addMember(?,?,?,?,?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setString(1, member.getFirstName());

st.setString(2, member.getLastName());

st.setString(3, member.getEmailAddress());

st.setString(4, member.getContactNumber());

st.setString(5, member.getBirthdate());

st.setString(6, member.getAddress());

st.setString(7, member.getSuburb());

st.setString(8, member.getCity());

st.setString(9, member.getPostalCode());

st.executeQuery();

}

/\*\*

\* Update the information of the member

\* @param member object to be updated

\* @throws Exception

\*/

public static void updateMember(Member member) throws Exception {

String sql = "call sp\_updateMember(?,?,?,?,?,?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setInt(1, member.getId());

st.setString(2, member.getFirstName());

st.setString(3, member.getLastName());

st.setString(4, member.getEmailAddress());

st.setString(5, member.getContactNumber());

st.setString(6, member.getBirthdate());

st.setString(7, member.getAddress());

st.setString(8, member.getSuburb());

st.setString(9, member.getCity());

st.setString(10, member.getPostalCode());

st.executeQuery();

}

/\*\*

\* Get the list of members

\* @return list of members

\* @throws Exception

\*/

public static ArrayList<Person> getAllMember() throws Exception {

ArrayList<Person> personList = new ArrayList();

String sql = "call sp\_searchMember()";

ResultSet rs = getStatement(sql).executeQuery();

while (rs.next()) {

Member member = new Member(rs.getInt("person\_id"), rs.getString("birthdate"), rs.getString("address"), rs.getString("suburb"), rs.getString("city"), rs.getString("postal\_code"), rs.getString("first\_name"), rs.getString("last\_name"), rs.getString("email\_address"), rs.getString("contact\_number"));

personList.add(member);

}

return personList;

}

/\*\*

\* Get the list of members filtered by the name

\* @param name filter

\* @return list of members

\* @throws Exception

\*/

public static ArrayList<Person> getMemberByName(String name) throws Exception {

ArrayList<Person> personList = new ArrayList();

String sql = "call sp\_searchMemberByName(?)";

CallableStatement st = getStatement(sql);

st.setString(1, name);

ResultSet rs = st.executeQuery();

while (rs.next()) {

Member member = new Member(rs.getInt("person\_id"), rs.getString("birthdate"), rs.getString("address"), rs.getString("suburb"), rs.getString("city"), rs.getString("postal\_code"), rs.getString("first\_name"), rs.getString("last\_name"), rs.getString("email\_address"), rs.getString("contact\_number"));

personList.add(member);

}

return personList;

}

/\*\*

\* Deactivation of the selected member

\* @param id

\* @throws Exception

\*/

public static void deactivateMember(int id) throws Exception {

String sql = "call sp\_deactivateMember(?)";

CallableStatement st = getStatement(sql);

st.setInt(1, id);

st.executeQuery();

}

/\*\*

\* Get the list of members filtered by name

\* @return list of members filtered by the name

\* @throws Exception

\*/

public static ArrayList<String> getAllMemberName() throws Exception {

ArrayList<String> personList = new ArrayList();

String sql = "call sp\_searchPerson()";

CallableStatement st = getStatement(sql);

ResultSet rs = st.executeQuery();

while (rs.next()) {

personList.add(rs.getString("name"));

}

return personList;

}

}

**UserDAL**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAL;

import Model.Person;

import Model.User;

import java.sql.CallableStatement;

import java.sql.ResultSet;

import java.util.ArrayList;

/\*\*

\*

\* @author massami

\*/

public class UserDAL extends BaseDAL {

/\*\*

\* Insert the user to the database

\* @param user object to be inserted

\* @throws Exception

\*/

public static void addUser(User user) throws Exception {

String sql = "call sp\_addUser(?,?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setString(1, user.getFirstName());

st.setString(2, user.getLastName());

st.setString(3, user.getEmailAddress());

st.setString(4, user.getContactNumber());

st.setString(5, user.getUsername());

st.setString(6, user.getPassword());

st.executeQuery();

}

/\*\*

\* Update the information of the user

\* @param user object to be updated

\* @throws Exception

\*/

public static void updateUser(User user) throws Exception {

String sql = "call sp\_updateUser(?,?,?,?,?,?)";

CallableStatement st = getStatement(sql);

st.setInt(1, user.getId());

st.setString(2, user.getFirstName());

st.setString(3, user.getLastName());

st.setString(4, user.getEmailAddress());

st.setString(5, user.getContactNumber());

st.setString(6, user.getUsername());

st.executeQuery();

}

/\*\*

\* Get list of users

\* @return list of users

\* @throws Exception

\*/

public static ArrayList<Person> getAllUser() throws Exception {

ArrayList<Person> personList = new ArrayList();

String sql = "call sp\_searchUser()";

ResultSet rs = getStatement(sql).executeQuery();

while (rs.next()) {

User user = new User(rs.getString("username"), rs.getInt("person\_id"), rs.getString("first\_name"), rs.getString("last\_name"), rs.getString("email\_address"), rs.getString("contact\_number"));

personList.add(user);

}

return personList;

}

/\*\*

\* Get list of users filtered by name

\* @param name filter

\* @return list of users

\* @throws Exception

\*/

public static ArrayList<Person> getUserByName(String name) throws Exception {

ArrayList<Person> personList = new ArrayList();

String sql = "call sp\_searchUserByName(?)";

CallableStatement st = getStatement(sql);

st.setString(1, name);

ResultSet rs = st.executeQuery();

while (rs.next()) {

User user = new User(rs.getString("username"), rs.getInt("person\_id"), rs.getString("first\_name"), rs.getString("last\_name"), rs.getString("email\_address"), rs.getString("contact\_number"));

personList.add(user);

}

return personList;

}

/\*\*

\* Deactivate the selected member

\* @param id

\* @throws Exception

\*/

public static void deactivateUser(int id) throws Exception {

String sql = "call sp\_deactivateUser(?)";

CallableStatement st = getStatement(sql);

st.setInt(1, id);

st.executeQuery();

}

/\*\*

\* Update the password of the user

\* @param username

\* @param password

\* @param newPassword

\* @throws Exception

\*/

public static void updatePassword(String username, String password, String newPassword) throws Exception {

String sql = "call sp\_updatePassword(?,?,?)";

CallableStatement st = getStatement(sql);

st.setString(1, username);

st.setString(2, password);

st.setString(3, newPassword);

st.executeQuery();

}

/\*\*

\* Check if the password is correct

\* @param username

\* @param password

\* @return

\* @throws Exception

\*/

public static ArrayList<Person> checkPassword(String username, String password) throws Exception {

ArrayList<Person> personList = new ArrayList();

String sql = "call sp\_checkPassword(?,?)";

CallableStatement st = getStatement(sql);

st.setString(1, username);

st.setString(2, password);

ResultSet rs = st.executeQuery();

while (rs.next()) {

User user = new User(rs.getString("username"));

personList.add(user);

}

return personList;

}

/\*\*

\* Reset the password of the member

\* @param id

\* @throws Exception

\*/

public static void resetPassword(int id) throws Exception {

String sql = "call sp\_resetPassword(?)";

CallableStatement st = getStatement(sql);

st.setInt(1, id);

st.executeQuery();

}

}

**GUI Package**

**ApplicationGUI**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import javax.swing.JInternalFrame;

/\*\*

\*

\* @author Rom

\*/

public class ApplicationGUI extends javax.swing.JFrame {

/\*\*

\* Creates new form MainScreenView

\*/

public ApplicationGUI() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

mainScreenPane = new javax.swing.JDesktopPane();

jMenuBar1 = new javax.swing.JMenuBar();

jMenu4 = new javax.swing.JMenu();

borrowManagementMenuItem = new javax.swing.JMenuItem();

jMenu2 = new javax.swing.JMenu();

bookManagementMenuItem = new javax.swing.JMenuItem();

dvdMenuItem = new javax.swing.JMenuItem();

jMenu3 = new javax.swing.JMenu();

memberManagementMenuItem = new javax.swing.JMenuItem();

jMenu1 = new javax.swing.JMenu();

changePasswordMenuItem = new javax.swing.JMenuItem();

userManagementMenuItem = new javax.swing.JMenuItem();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setTitle("Library System");

setSize(new java.awt.Dimension(1366, 768));

mainScreenPane.setMinimumSize(new java.awt.Dimension(1366, 768));

javax.swing.GroupLayout mainScreenPaneLayout = new javax.swing.GroupLayout(mainScreenPane);

mainScreenPane.setLayout(mainScreenPaneLayout);

mainScreenPaneLayout.setHorizontalGroup(

mainScreenPaneLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGap(0, 1366, Short.MAX\_VALUE)

);

mainScreenPaneLayout.setVerticalGroup(

mainScreenPaneLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGap(0, 768, Short.MAX\_VALUE)

);

getContentPane().add(mainScreenPane, java.awt.BorderLayout.CENTER);

jMenu4.setText("Borrow");

jMenu4.setFont(new java.awt.Font("Dialog", 1, 18)); // NOI18N

borrowManagementMenuItem.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

borrowManagementMenuItem.setText("Borrow Management");

borrowManagementMenuItem.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

borrowManagementMenuItemActionPerformed(evt);

}

});

jMenu4.add(borrowManagementMenuItem);

jMenuBar1.add(jMenu4);

jMenu2.setText("Catalogue");

jMenu2.setFont(new java.awt.Font("Dialog", 1, 18)); // NOI18N

bookManagementMenuItem.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

bookManagementMenuItem.setText("Books");

bookManagementMenuItem.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

bookManagementMenuItemActionPerformed(evt);

}

});

jMenu2.add(bookManagementMenuItem);

dvdMenuItem.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

dvdMenuItem.setText("Dvds");

dvdMenuItem.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

dvdMenuItemActionPerformed(evt);

}

});

jMenu2.add(dvdMenuItem);

jMenuBar1.add(jMenu2);

jMenu3.setText("Members");

jMenu3.setFont(new java.awt.Font("Dialog", 1, 18)); // NOI18N

memberManagementMenuItem.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

memberManagementMenuItem.setText("Member Management");

memberManagementMenuItem.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

memberManagementMenuItemActionPerformed(evt);

}

});

jMenu3.add(memberManagementMenuItem);

jMenuBar1.add(jMenu3);

jMenu1.setText("Users");

jMenu1.setFont(new java.awt.Font("Dialog", 1, 18)); // NOI18N

jMenu1.setMargin(new java.awt.Insets(10, 10, 10, 10));

changePasswordMenuItem.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

changePasswordMenuItem.setText("Change Password");

changePasswordMenuItem.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

changePasswordMenuItemActionPerformed(evt);

}

});

jMenu1.add(changePasswordMenuItem);

userManagementMenuItem.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

userManagementMenuItem.setText("User Management");

userManagementMenuItem.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

userManagementMenuItemActionPerformed(evt);

}

});

jMenu1.add(userManagementMenuItem);

jMenuBar1.add(jMenu1);

setJMenuBar(jMenuBar1);

pack();

}// </editor-fold>

private void bookManagementMenuItemActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

getIFrame(new BookManagementIFrame());

}

private void memberManagementMenuItemActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

getIFrame(new MemberManagementIFrame());

}

private void userManagementMenuItemActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

getIFrame(new UserManagementIFrame());

}

private void changePasswordMenuItemActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

getIFrame(new ChangePasswordIFrame());

}

private void dvdMenuItemActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

getIFrame(new DvdManagementIFrame());

}

private void borrowManagementMenuItemActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

getIFrame(new BorrowManagementIFrame());

}

/\*\*

\* Display frame to the application

\*

\* @param frame to be displayed

\*/

private void getIFrame(JInternalFrame frame) {

if (frame != null) {

frame.dispose();

}

mainScreenPane.add(frame);

frame.setVisible(true);

}

/\*\*

\* @param args the command line arguments

\*/

// public static void main(String args[]) {

// /\* Set the Nimbus look and feel \*/

// //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

// /\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

// \* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

// \*/

// try {

// for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

// if ("Nimbus".equals(info.getName())) {

// javax.swing.UIManager.setLookAndFeel(info.getClassName());

// break;

// }

// }

// } catch (ClassNotFoundException ex) {

// java.util.logging.Logger.getLogger(ApplicationGUI.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

// } catch (InstantiationException ex) {

// java.util.logging.Logger.getLogger(ApplicationGUI.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

// } catch (IllegalAccessException ex) {

// java.util.logging.Logger.getLogger(ApplicationGUI.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

// } catch (javax.swing.UnsupportedLookAndFeelException ex) {

// java.util.logging.Logger.getLogger(ApplicationGUI.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

// }

// //</editor-fold>

// //</editor-fold>

// //</editor-fold>

// //</editor-fold>

//

// /\* Create and display the form \*/

// java.awt.EventQueue.invokeLater(new Runnable() {

// public void run() {

// new ApplicationGUI().setVisible(true);

// }

// });

// }

// Variables declaration - do not modify

private javax.swing.JMenuItem bookManagementMenuItem;

private javax.swing.JMenuItem borrowManagementMenuItem;

private javax.swing.JMenuItem changePasswordMenuItem;

private javax.swing.JMenuItem dvdMenuItem;

private javax.swing.JMenu jMenu1;

private javax.swing.JMenu jMenu2;

private javax.swing.JMenu jMenu3;

private javax.swing.JMenu jMenu4;

private javax.swing.JMenuBar jMenuBar1;

private javax.swing.JDesktopPane mainScreenPane;

private javax.swing.JMenuItem memberManagementMenuItem;

private javax.swing.JMenuItem userManagementMenuItem;

// End of variables declaration

}

**BookManagementIFrame**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.Book;

import Model.Catalogue;

import Utility.GUIUtility;

import java.util.ArrayList;

import java.util.List;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableModel;

/\*\*

\*

\* @author Rom

\*/

public class BookManagementIFrame extends javax.swing.JInternalFrame {

List<Catalogue> listCatalogue = new ArrayList();

Book selectedBook = new Book();

/\*\*

\* Creates new form BooKInternalFrame

\*/

public BookManagementIFrame() {

initComponents();

//updateButton.setEnabled(false);

//deleteButton.setEnabled(false);

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel2 = new javax.swing.JPanel();

jScrollPane1 = new javax.swing.JScrollPane();

bookTable = new javax.swing.JTable();

bookTextField = new javax.swing.JTextField();

searchButton = new javax.swing.JButton();

bookPanel = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

titleTextField = new javax.swing.JTextField();

jLabel2 = new javax.swing.JLabel();

descriptionTextField = new javax.swing.JTextField();

jLabel3 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

languageTextField = new javax.swing.JTextField();

jLabel5 = new javax.swing.JLabel();

authorTextField = new javax.swing.JTextField();

jLabel6 = new javax.swing.JLabel();

genreTextField = new javax.swing.JTextField();

jLabel7 = new javax.swing.JLabel();

isbnTextField = new javax.swing.JTextField();

jLabel8 = new javax.swing.JLabel();

jLabel9 = new javax.swing.JLabel();

noteTextField = new javax.swing.JTextField();

jPanel3 = new javax.swing.JPanel();

addButton = new javax.swing.JButton();

updateButton = new javax.swing.JButton();

deleteButton = new javax.swing.JButton();

statusComboBox = new javax.swing.JComboBox<>();

releaseDatePicker = new org.jdesktop.swingx.JXDatePicker();

setClosable(true);

setTitle("Book Management");

setMaximumSize(new java.awt.Dimension(1180, 580));

setMinimumSize(new java.awt.Dimension(1180, 580));

setPreferredSize(new java.awt.Dimension(1180, 580));

bookTable.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"Title", "Author", "Description", "ISBN"

}

) {

Class[] types = new Class [] {

java.lang.String.class, java.lang.Object.class, java.lang.Object.class, java.lang.Object.class

};

public Class getColumnClass(int columnIndex) {

return types [columnIndex];

}

});

bookTable.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

bookTableMouseClicked(evt);

}

});

jScrollPane1.setViewportView(bookTable);

searchButton.setText("Search");

searchButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

searchButtonActionPerformed(evt);

}

});

bookPanel.setBorder(javax.swing.BorderFactory.createTitledBorder(null, "Book Details", javax.swing.border.TitledBorder.DEFAULT\_JUSTIFICATION, javax.swing.border.TitledBorder.DEFAULT\_POSITION, new java.awt.Font("Dialog", 1, 18))); // NOI18N

bookPanel.setName(""); // NOI18N

bookPanel.setPreferredSize(new java.awt.Dimension(520, 540));

jLabel1.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel1.setText("Title");

titleTextField.setName("title"); // NOI18N

jLabel2.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel2.setText("Description");

descriptionTextField.setName("description"); // NOI18N

jLabel3.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel3.setText("Release Date");

jLabel4.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel4.setText("Language");

languageTextField.setName("language"); // NOI18N

jLabel5.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel5.setText("Author");

authorTextField.setName("author"); // NOI18N

jLabel6.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel6.setText("Genre");

genreTextField.setName("genre"); // NOI18N

jLabel7.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel7.setText("ISBN");

isbnTextField.setName("isbn"); // NOI18N

jLabel8.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel8.setText("Status");

jLabel9.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel9.setText("Note");

noteTextField.setName(""); // NOI18N

jPanel3.setMinimumSize(new java.awt.Dimension(160, 80));

jPanel3.setPreferredSize(new java.awt.Dimension(264, 80));

jPanel3.setLayout(new java.awt.GridLayout(2, 2));

addButton.setText("Add New");

addButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

addButtonActionPerformed(evt);

}

});

jPanel3.add(addButton);

updateButton.setText("Update");

updateButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

updateButtonActionPerformed(evt);

}

});

jPanel3.add(updateButton);

deleteButton.setText("Delete");

deleteButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

deleteButtonActionPerformed(evt);

}

});

jPanel3.add(deleteButton);

statusComboBox.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Available", "Borrowed", "Under Maintenance", "Not in Stock" }));

statusComboBox.setName("status"); // NOI18N

releaseDatePicker.setName("releaseDate"); // NOI18N

javax.swing.GroupLayout bookPanelLayout = new javax.swing.GroupLayout(bookPanel);

bookPanel.setLayout(bookPanelLayout);

bookPanelLayout.setHorizontalGroup(

bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, bookPanelLayout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(bookPanelLayout.createSequentialGroup()

.addComponent(jLabel7)

.addGap(78, 78, 78)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(languageTextField)

.addComponent(isbnTextField)))

.addGroup(bookPanelLayout.createSequentialGroup()

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel4)

.addComponent(jLabel5)

.addComponent(jLabel6))

.addGap(42, 42, 42)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(authorTextField)

.addComponent(genreTextField)))

.addComponent(jPanel3, javax.swing.GroupLayout.DEFAULT\_SIZE, 378, Short.MAX\_VALUE)

.addGroup(bookPanelLayout.createSequentialGroup()

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel8)

.addComponent(jLabel9))

.addGap(67, 67, 67)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(noteTextField)

.addComponent(statusComboBox, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)))

.addGroup(bookPanelLayout.createSequentialGroup()

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel2)

.addComponent(jLabel1)

.addComponent(jLabel3))

.addGap(18, 18, 18)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(titleTextField, javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(descriptionTextField, javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(releaseDatePicker, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))))

.addGap(20, 20, 20))

);

bookPanelLayout.setVerticalGroup(

bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(bookPanelLayout.createSequentialGroup()

.addGap(30, 30, 30)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(titleTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2)

.addComponent(descriptionTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel3)

.addComponent(releaseDatePicker, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel8)

.addComponent(statusComboBox, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(noteTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel9))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(languageTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGroup(bookPanelLayout.createSequentialGroup()

.addComponent(jLabel4)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel5)

.addComponent(authorTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))))

.addGap(7, 7, 7)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(genreTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel6))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(bookPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(isbnTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel7))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 104, Short.MAX\_VALUE)

.addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 80, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(bookTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 361, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(searchButton))

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 686, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addComponent(bookPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 430, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(20, 20, 20))

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addContainerGap()

.addComponent(bookPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 500, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(javax.swing.GroupLayout.Alignment.LEADING, jPanel2Layout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(bookTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(searchButton))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 442, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGap(46, 46, 46))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

pack();

}// </editor-fold>

private void bookTableMouseClicked(java.awt.event.MouseEvent evt) {

try {

//Get the current selected table row

int i = bookTable.getSelectedRow();

//Get the selected book item

selectedBook = (Book) listCatalogue.get(i);

//Add data to the panel controls

setDataToComponent(selectedBook);

//Set control availability

updateButton.setEnabled(true);

deleteButton.setEnabled(true);

addButton.setEnabled(false);

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void searchButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Get the model of the table

DefaultTableModel model = (DefaultTableModel) bookTable.getModel();

//Clean the table content

model.setRowCount(0);

//Initialize rows

Object[] rows = new Object[4];

//Initialize Book object

Catalogue catalogue = new Book();

//Search the requested book

listCatalogue = bookTextField.getText().equals("") ? catalogue.Search() : catalogue.SearchByTitle(bookTextField.getText());

if (listCatalogue.size() > 0) {

//Add book data to the table

for (Catalogue c : listCatalogue) {

Book b = (Book) c;

rows[0] = b.getTitle();

rows[1] = b.getAuthor();

rows[2] = b.getDescription();

rows[3] = b.getIsbn();

model.addRow(rows);

}

}else{

JOptionPane.showMessageDialog(this, "No book found, please include a book first");

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void addButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Check if input data is correct

String msg = GUIUtility.validateInput(bookPanel);

if (msg.length() == 0) {

//Load book object with the panel data

Catalogue catalogue = getComponentData(new Book());

//Insert book into the database

catalogue.Add();

//Displays message

JOptionPane.showMessageDialog(this, "Book registered successfully");

//Clean data in controls

GUIUtility.cleanComponents(bookPanel);

//Reset the table`s content

GUIUtility.cleanTable(bookTable);

} else {

JOptionPane.showMessageDialog(this, msg);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void updateButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(bookPanel);

if (msg.length() == 0) {

//Get the selected row in the table

int i = bookTable.getSelectedRow();

//Get the title of the selected item

String title = bookTable.getModel().getValueAt(i, 0).toString();

//Get the selected book

Book book = (Book) listCatalogue.stream().filter(p -> p.getTitle().equals(title)).findFirst().get();

//Get components data

Catalogue catalogue = getComponentData(book);

//Update book data

catalogue.Update();

//Display message

JOptionPane.showMessageDialog(this, "Book updated successfully");

//Clean data in controls

GUIUtility.cleanComponents(bookPanel);

//Reset the table`s content

GUIUtility.cleanTable(bookTable);

//Enable buttons

addButton.setEnabled(true);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void deleteButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(bookPanel);

if (msg.length() == 0) {

//Get the selected row in the table

int i = bookTable.getSelectedRow();

//Get the current table data

TableModel model = bookTable.getModel();

//Get the title of the selected item

String title = model.getValueAt(i, 0).toString();

//Initialize objects

Catalogue catalogue = new Book();

//Filter book based on title

int id = ((Book) (listCatalogue.stream().filter(p -> p.getTitle().equals(title)).findFirst().get())).getId();

//Update book data

catalogue.Delete(id);

//Display message

JOptionPane.showMessageDialog(this, "Book deleted successfully");

//Clean data in controls

GUIUtility.cleanComponents(bookPanel);

//Enable buttons

addButton.setEnabled(true);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

/\*\*

\* Get the data from the components

\*

\* @param book object to be filled

\* @return filled object

\* @throws Exception

\*/

private Book getComponentData(Book book) throws Exception {

book.setAuthor(authorTextField.getText());

book.setGenre(genreTextField.getText());

book.setIsbn(isbnTextField.getText());

book.setTitle(titleTextField.getText());

book.setDescription(descriptionTextField.getText());

book.setReleaseDate(GUIUtility.convertDateToString(releaseDatePicker.getDate()));

book.setStatus(statusComboBox.getSelectedItem().toString());

book.setNote(noteTextField.getText());

book.setLanguage(languageTextField.getText());

return book;

}

/\*\*

\* Set data from the object to the components of the GUI

\*

\* @param book get data from

\* @throws Exception

\*/

private void setDataToComponent(Book book) throws Exception {

titleTextField.setText(book.getTitle());

authorTextField.setText(book.getAuthor());

descriptionTextField.setText(book.getDescription());

isbnTextField.setText(book.getIsbn());

genreTextField.setText(book.getGenre());

noteTextField.setText(book.getNote());

languageTextField.setText(book.getLanguage());

releaseDatePicker.setDate(GUIUtility.convertToDate(book.getReleaseDate()));

statusComboBox.setSelectedItem(book.getStatus());

}

// Variables declaration - do not modify

private javax.swing.JButton addButton;

private javax.swing.JTextField authorTextField;

private javax.swing.JPanel bookPanel;

private javax.swing.JTable bookTable;

private javax.swing.JTextField bookTextField;

private javax.swing.JButton deleteButton;

private javax.swing.JTextField descriptionTextField;

private javax.swing.JTextField genreTextField;

private javax.swing.JTextField isbnTextField;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JLabel jLabel8;

private javax.swing.JLabel jLabel9;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTextField languageTextField;

private javax.swing.JTextField noteTextField;

private org.jdesktop.swingx.JXDatePicker releaseDatePicker;

private javax.swing.JButton searchButton;

private javax.swing.JComboBox<String> statusComboBox;

private javax.swing.JTextField titleTextField;

private javax.swing.JButton updateButton;

// End of variables declaration

}

**BookSearchFrame**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.Book;

import Model.Catalogue;

import java.util.ArrayList;

import java.util.List;

import javax.swing.JInternalFrame;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

/\*\*

\*

\* @author massami

\*/

public class BookSearchFrame extends javax.swing.JFrame {

List<Catalogue> listCatalogue = new ArrayList();

Book selectedBook = new Book();

BorrowManagementIFrame frame;

/\*\*

\* Creates new form BookSearchFrame

\*/

public BookSearchFrame() {

initComponents();

}

public BookSearchFrame(JInternalFrame internal){

initComponents();

frame = (BorrowManagementIFrame) internal;

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

bookTextField = new javax.swing.JTextField();

searchButton = new javax.swing.JButton();

jScrollPane1 = new javax.swing.JScrollPane();

bookTable = new javax.swing.JTable();

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE\_ON\_CLOSE);

jLabel1.setText("Search: ");

searchButton.setText("Search");

searchButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

searchButtonActionPerformed(evt);

}

});

bookTable.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"Title", "Author", "Description", "ISBN"

}

) {

Class[] types = new Class [] {

java.lang.String.class, java.lang.Object.class, java.lang.Object.class, java.lang.Object.class

};

public Class getColumnClass(int columnIndex) {

return types [columnIndex];

}

});

bookTable.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

bookTableMouseClicked(evt);

}

});

jScrollPane1.setViewportView(bookTable);

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(bookTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 365, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(searchButton, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 571, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(0, 18, Short.MAX\_VALUE))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(bookTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(searchButton))

.addGap(18, 18, 18)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 442, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(17, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGap(0, 0, 0))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void bookTableMouseClicked(java.awt.event.MouseEvent evt) {

try {

//Get the current selected table row

int i = bookTable.getSelectedRow();

//Get the selected book item

selectedBook = (Book) listCatalogue.get(i);

//Add selected book to table

frame.addItem(selectedBook);

//Close Book search window

this.dispose();

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void searchButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Get the model of the table

DefaultTableModel model = (DefaultTableModel) bookTable.getModel();

//Clean the table content

model.setRowCount(0);

//Initialize rows

Object[] rows = new Object[4];

//Initialize Book object

Catalogue catalogue = new Book();

//Search the requested book

listCatalogue = bookTextField.getText().equals("") ? catalogue.SearchAvailableItems(): catalogue.SearchAvailableItemsByTitle(bookTextField.getText());

//Add book data to the table

for (Catalogue c : listCatalogue) {

Book b = (Book) c;

rows[0] = b.getTitle();

rows[1] = b.getAuthor();

rows[2] = b.getDescription();

rows[3] = b.getIsbn();

model.addRow(rows);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(BookSearchFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(BookSearchFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(BookSearchFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(BookSearchFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new BookSearchFrame().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JTable bookTable;

private javax.swing.JTextField bookTextField;

private javax.swing.JLabel jLabel1;

private javax.swing.JPanel jPanel1;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JButton searchButton;

// End of variables declaration

}

**BorrowManagementIFrame**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.Book;

import Model.Borrow;

import Model.Catalogue;

import Model.Member;

import Utility.GUIUtility;

import java.util.ArrayList;

import java.util.Date;

import java.util.List;

import javax.swing.DefaultComboBoxModel;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

/\*\*

\*

\* @author Rom

\*/

public class BorrowManagementIFrame extends javax.swing.JInternalFrame {

List<Catalogue> listCatalogue = new ArrayList();

ArrayList<String> memberList = new ArrayList();

/\*\*

\* Creates new form BorrowManagementIFrame

\*/

public BorrowManagementIFrame() {

initComponents();

LoadComboBox();

LoadTable();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel2 = new javax.swing.JPanel();

jScrollPane1 = new javax.swing.JScrollPane();

borrowTable = new javax.swing.JTable();

borrowPanel = new javax.swing.JPanel();

jLabel10 = new javax.swing.JLabel();

jPanel3 = new javax.swing.JPanel();

borrowButton = new javax.swing.JButton();

returnButton = new javax.swing.JButton();

dueDatePicker = new org.jdesktop.swingx.JXDatePicker();

jLabel12 = new javax.swing.JLabel();

titleTextField = new javax.swing.JTextField();

jLabel13 = new javax.swing.JLabel();

IdTextField = new javax.swing.JTextField();

memberComboBox = new javax.swing.JComboBox<>();

addBookButton = new javax.swing.JButton();

removeBookButton = new javax.swing.JButton();

jLabel3 = new javax.swing.JLabel();

addDvdButton = new javax.swing.JButton();

setClosable(true);

setTitle("Borrow Management");

setMaximumSize(new java.awt.Dimension(1176, 420));

setMinimumSize(new java.awt.Dimension(1176, 420));

borrowTable.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"Title", "Description", "Status"

}

) {

Class[] types = new Class [] {

java.lang.String.class, java.lang.Object.class, java.lang.Object.class

};

boolean[] canEdit = new boolean [] {

false, false, true

};

public Class getColumnClass(int columnIndex) {

return types [columnIndex];

}

public boolean isCellEditable(int rowIndex, int columnIndex) {

return canEdit [columnIndex];

}

});

borrowTable.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

borrowTableMouseClicked(evt);

}

});

jScrollPane1.setViewportView(borrowTable);

borrowPanel.setBorder(javax.swing.BorderFactory.createTitledBorder(null, "Borrow Details", javax.swing.border.TitledBorder.DEFAULT\_JUSTIFICATION, javax.swing.border.TitledBorder.DEFAULT\_POSITION, new java.awt.Font("Dialog", 1, 18))); // NOI18N

borrowPanel.setMaximumSize(new java.awt.Dimension(430, 310));

borrowPanel.setMinimumSize(new java.awt.Dimension(430, 310));

borrowPanel.setName(""); // NOI18N

borrowPanel.setPreferredSize(new java.awt.Dimension(430, 310));

jLabel10.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel10.setText("Due Date");

jPanel3.setMaximumSize(new java.awt.Dimension(380, 80));

jPanel3.setMinimumSize(new java.awt.Dimension(380, 80));

jPanel3.setPreferredSize(new java.awt.Dimension(380, 80));

jPanel3.setLayout(new java.awt.GridLayout(1, 2));

borrowButton.setText("Borrow Item");

borrowButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

borrowButtonActionPerformed(evt);

}

});

jPanel3.add(borrowButton);

returnButton.setText("Return Item");

returnButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

returnButtonActionPerformed(evt);

}

});

jPanel3.add(returnButton);

dueDatePicker.setName("dueDate"); // NOI18N

jLabel12.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel12.setText("Title");

titleTextField.setEnabled(false);

jLabel13.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel13.setText("Id");

IdTextField.setEditable(false);

javax.swing.GroupLayout borrowPanelLayout = new javax.swing.GroupLayout(borrowPanel);

borrowPanel.setLayout(borrowPanelLayout);

borrowPanelLayout.setHorizontalGroup(

borrowPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(borrowPanelLayout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(borrowPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel3, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGroup(borrowPanelLayout.createSequentialGroup()

.addGroup(borrowPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel10)

.addComponent(jLabel12)

.addComponent(jLabel13))

.addGap(38, 38, 38)

.addGroup(borrowPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(dueDatePicker, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(titleTextField)

.addComponent(IdTextField))))

.addGap(20, 20, 20))

);

borrowPanelLayout.setVerticalGroup(

borrowPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(borrowPanelLayout.createSequentialGroup()

.addGap(22, 22, 22)

.addGroup(borrowPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jLabel13)

.addComponent(IdTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(borrowPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jLabel12)

.addComponent(titleTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(borrowPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jLabel10)

.addComponent(dueDatePicker, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 124, Short.MAX\_VALUE)

.addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 36, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

memberComboBox.addItemListener(new java.awt.event.ItemListener() {

public void itemStateChanged(java.awt.event.ItemEvent evt) {

memberComboBoxItemStateChanged(evt);

}

});

addBookButton.setText("Add Book");

addBookButton.setPreferredSize(new java.awt.Dimension(100, 40));

addBookButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

addBookButtonActionPerformed(evt);

}

});

removeBookButton.setText("Remove");

removeBookButton.setPreferredSize(new java.awt.Dimension(100, 40));

removeBookButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

removeBookButtonActionPerformed(evt);

}

});

jLabel3.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel3.setText("Library Member");

addDvdButton.setText("Add DVD");

addDvdButton.setPreferredSize(new java.awt.Dimension(100, 40));

addDvdButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

addDvdButtonActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(jLabel3)

.addGap(18, 18, 18)

.addComponent(memberComboBox, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(addDvdButton, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(addBookButton, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(removeBookButton, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(8, 8, 8))

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 686, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGap(18, 18, 18)

.addComponent(borrowPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(20, Short.MAX\_VALUE))

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(memberComboBox, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel3))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 255, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addComponent(borrowPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(addBookButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 23, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(addDvdButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 23, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(removeBookButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 23, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void borrowButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(borrowPanel);

if (msg.length() == 0) {

if (borrowTable.getSelectedRow() >= 0) {

//Initialize new object

Borrow borrow = new Borrow();

//Get the member id

int memberId = borrow.searchMemberId(memberComboBox.getSelectedItem().toString());

//Get the id of the item

int catalogueId = Integer.parseInt(IdTextField.getText());

//Find the selected item

Catalogue selectedItem = listCatalogue.stream().filter(p -> p.getId() == catalogueId).findFirst().get();

if (selectedItem.getStatus().equals("Available")) {

//Add items from the list

borrow.setBorrowDate(GUIUtility.convertDateToString(new Date()));

borrow.setDueDate(GUIUtility.convertDateToString(dueDatePicker.getDate()));

borrow.setMemberId(memberId);

borrow.setCatalogueId(selectedItem.getId());

borrow.addBorrow(borrow);

//Display message

JOptionPane.showMessageDialog(this, "Item borrowed successfully");

//Update the status

listCatalogue.stream().filter(p -> p.getId() == catalogueId).findFirst().get().setStatus("Borrowed");

//Clean components

GUIUtility.cleanComponents(borrowPanel);

//Update table data

updateTable();

} else {

JOptionPane.showMessageDialog(this, "The selected item is not available");

}

} else {

JOptionPane.showMessageDialog(this, "Please select an item");

}

} else {

JOptionPane.showMessageDialog(this, msg);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void addBookButtonActionPerformed(java.awt.event.ActionEvent evt) {

BookSearchFrame frame = new BookSearchFrame(this);

frame.setVisible(true);

}

private void removeBookButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Get the current selected table row

int i = borrowTable.getSelectedRow();

//Get the selected book item

Book selectedBook = (Book) listCatalogue.get(i);

//remove selected book from collection

listCatalogue.remove(selectedBook);

//Update the table content

updateTable();

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void addDvdButtonActionPerformed(java.awt.event.ActionEvent evt) {

DvdSearchFrame frame = new DvdSearchFrame(this);

frame.setVisible(true);

}

private void borrowTableMouseClicked(java.awt.event.MouseEvent evt) {

try {

//Get the current selected table row

int i = borrowTable.getSelectedRow();

//Get the selected book item

Catalogue c = listCatalogue.get(i);

//Add data to the panel controls

setDataToComponent(c);

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void returnButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//No items is selected

if (borrowTable.getSelectedRow() >= 0) {

//Initialize new object

Borrow borrow = new Borrow();

//Get the member id

int memberId = borrow.searchMemberId(memberComboBox.getSelectedItem().toString());

//Get the id of the item

int catalogueId = Integer.parseInt(IdTextField.getText());

//Find the selected item

Catalogue selectedItem = listCatalogue.stream().filter(p -> p.getId() == catalogueId).findFirst().get();

if (selectedItem.getStatus().equals("Borrowed")) {

//Add items from the list

borrow.setBorrowDate(GUIUtility.convertDateToString(new Date()));

borrow.setMemberId(memberId);

borrow.setCatalogueId(selectedItem.getId());

borrow.returnItem(borrow);

//Display message

JOptionPane.showMessageDialog(this, "Item returned successfully");

//Update the status

listCatalogue.stream().filter(p -> p.getId() == catalogueId).findFirst().get().setStatus("Available");

//Clean components

GUIUtility.cleanComponents(borrowPanel);

//Update table data

updateTable();

} else {

JOptionPane.showMessageDialog(this, "The item is available to borrow");

}

} else {

JOptionPane.showMessageDialog(this, "Please select an item");

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void memberComboBoxItemStateChanged(java.awt.event.ItemEvent evt) {

try {

//Load the table with the items of the member

LoadTable();

//update the items of the table

updateTable();

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

/\*\*

\* Load data to the components of the panel

\*

\* @param c

\*/

private void setDataToComponent(Catalogue c) {

IdTextField.setText(String.valueOf(c.getId()));

titleTextField.setText(c.getTitle());

}

/\*\*

\* Load ComboBox with the members

\*/

private void LoadComboBox() {

try {

Member member = new Member();

memberList = member.SearchMemberName();

memberComboBox.setModel((new DefaultComboBoxModel(memberList.toArray())));

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

/\*\*

\* Update table with the catalogue items

\*/

private void updateTable() {

//Get the model of the table

DefaultTableModel model = (DefaultTableModel) borrowTable.getModel();

//Clean the table content

model.setRowCount(0);

//Initialize rows

Object[] rows = new Object[3];

//Add book data to the table

for (Catalogue c : listCatalogue) {

rows[0] = c.getTitle();

rows[1] = c.getDescription();

rows[2] = c.getStatus();

model.addRow(rows);

}

}

/\*\*

\* Add item to the list and update the table

\*

\* @param cat

\*/

public void addItem(Catalogue cat) {

try {

//Add new book to the catalogue list

if (!listCatalogue.contains(cat)) {

listCatalogue.add(cat);

}

//Update table content

updateTable();

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

/\*\*

\* Load the table with items of the member

\*/

public void LoadTable() {

try {

//Initialize object

Borrow borrow = new Borrow();

//Search the transaction book of the user

listCatalogue = borrow.searchBorrowedItem(memberComboBox.getSelectedItem().toString());

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

// Variables declaration - do not modify

private javax.swing.JTextField IdTextField;

private javax.swing.JButton addBookButton;

private javax.swing.JButton addDvdButton;

private javax.swing.JButton borrowButton;

private javax.swing.JPanel borrowPanel;

private javax.swing.JTable borrowTable;

private org.jdesktop.swingx.JXDatePicker dueDatePicker;

private javax.swing.JLabel jLabel10;

private javax.swing.JLabel jLabel12;

private javax.swing.JLabel jLabel13;

private javax.swing.JLabel jLabel3;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JComboBox<String> memberComboBox;

private javax.swing.JButton removeBookButton;

private javax.swing.JButton returnButton;

private javax.swing.JTextField titleTextField;

// End of variables declaration

}

**ChangePasswordIFrame**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.Person;

import Model.User;

import Shared.SharedData;

import Utility.GUIUtility;

import java.util.ArrayList;

import javax.swing.JOptionPane;

/\*\*

\*

\* @author Rom

\*/

public class ChangePasswordIFrame extends javax.swing.JInternalFrame {

/\*\*

\* Creates new form ChangePasswordIFrame

\*/

public ChangePasswordIFrame() {

initComponents();

getUsername();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

changePasswordPanel = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

passwordTextField = new javax.swing.JTextField();

jLabel8 = new javax.swing.JLabel();

confirmPasswordTextField = new javax.swing.JTextField();

newPasswordTextField = new javax.swing.JTextField();

jLabel10 = new javax.swing.JLabel();

jPanel3 = new javax.swing.JPanel();

changePasswordButton = new javax.swing.JButton();

cancelButton = new javax.swing.JButton();

usernameLabel = new javax.swing.JLabel();

setClosable(true);

setTitle("Change Password");

changePasswordPanel.setBorder(javax.swing.BorderFactory.createTitledBorder(null, "Change Password", javax.swing.border.TitledBorder.DEFAULT\_JUSTIFICATION, javax.swing.border.TitledBorder.DEFAULT\_POSITION, new java.awt.Font("Dialog", 1, 18))); // NOI18N

changePasswordPanel.setName(""); // NOI18N

changePasswordPanel.setPreferredSize(new java.awt.Dimension(520, 540));

jLabel1.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel1.setText("Username");

jLabel2.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel2.setText("Password");

jLabel8.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel8.setText("Confirm Password");

jLabel10.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel10.setText("New Password");

jPanel3.setPreferredSize(new java.awt.Dimension(268, 40));

jPanel3.setLayout(new java.awt.GridLayout(1, 2));

changePasswordButton.setText("Change Password");

changePasswordButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

changePasswordButtonActionPerformed(evt);

}

});

jPanel3.add(changePasswordButton);

cancelButton.setText("Cancel");

cancelButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

cancelButtonActionPerformed(evt);

}

});

jPanel3.add(cancelButton);

usernameLabel.setText("jLabel3");

javax.swing.GroupLayout changePasswordPanelLayout = new javax.swing.GroupLayout(changePasswordPanel);

changePasswordPanel.setLayout(changePasswordPanelLayout);

changePasswordPanelLayout.setHorizontalGroup(

changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(changePasswordPanelLayout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel3, javax.swing.GroupLayout.DEFAULT\_SIZE, 378, Short.MAX\_VALUE)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, changePasswordPanelLayout.createSequentialGroup()

.addGroup(changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel2)

.addComponent(jLabel1))

.addGap(85, 85, 85)

.addGroup(changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(passwordTextField, javax.swing.GroupLayout.DEFAULT\_SIZE, 224, Short.MAX\_VALUE)

.addGroup(changePasswordPanelLayout.createSequentialGroup()

.addComponent(usernameLabel)

.addGap(0, 0, Short.MAX\_VALUE))))

.addGroup(changePasswordPanelLayout.createSequentialGroup()

.addGroup(changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel10)

.addComponent(jLabel8))

.addGap(29, 29, 29)

.addGroup(changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(confirmPasswordTextField)

.addComponent(newPasswordTextField))))

.addGap(20, 20, 20))

);

changePasswordPanelLayout.setVerticalGroup(

changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(changePasswordPanelLayout.createSequentialGroup()

.addGap(30, 30, 30)

.addGroup(changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(usernameLabel))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2)

.addComponent(passwordTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addGroup(changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel10)

.addComponent(newPasswordTextField, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(changePasswordPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel8)

.addComponent(confirmPasswordTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 80, Short.MAX\_VALUE)

.addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGap(0, 450, Short.MAX\_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(changePasswordPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 430, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGap(0, 366, Short.MAX\_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(changePasswordPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 305, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(46, Short.MAX\_VALUE)))

);

pack();

}// </editor-fold>

private void changePasswordButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

if (checkUserPassword() && isPasswordMatched()) {

//Get user information

String username = usernameLabel.getText();

String password = passwordTextField.getText();

String newPassword = newPasswordTextField.getText();

//Get components data

User user = new User();

//Update book data

user.UpdatePassword(username,password, newPassword);

//Display message

JOptionPane.showMessageDialog(this, "Password updated successfully");

//Clean the values of the components

GUIUtility.cleanComponents(changePasswordPanel);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void cancelButtonActionPerformed(java.awt.event.ActionEvent evt) {

GUIUtility.cleanComponents(changePasswordPanel);

}

/\*\*

\* Check if the new password matches

\*

\* @return

\*/

private boolean isPasswordMatched() {

if (!newPasswordTextField.getText().equals(confirmPasswordTextField.getText()) ||

newPasswordTextField.getText().equals("") &&

confirmPasswordTextField.getText().equals("")) {

JOptionPane.showMessageDialog(this, "Password does not match");

return false;

}

return true;

}

/\*\*

\* Verify if the password is correct

\* @return statement

\*/

private boolean checkUserPassword() {

try {

//Get components data

User user = new User();

//Get the user data

String username = usernameLabel.getText();

String password = passwordTextField.getText();

//Update book data

ArrayList<Person> p = user.CheckPassword(username,password);

if(p.isEmpty()){

JOptionPane.showMessageDialog(this, "Password is not correct");

return false;

}

else {

return true;

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, "Password is not correct");

return false;

}

}

/\*\*

\* Get the username of the member

\*/

private void getUsername(){

usernameLabel.setText(SharedData.username);

}

// Variables declaration - do not modify

private javax.swing.JButton cancelButton;

private javax.swing.JButton changePasswordButton;

private javax.swing.JPanel changePasswordPanel;

private javax.swing.JTextField confirmPasswordTextField;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel10;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel8;

private javax.swing.JPanel jPanel3;

private javax.swing.JTextField newPasswordTextField;

private javax.swing.JTextField passwordTextField;

private javax.swing.JLabel usernameLabel;

// End of variables declaration

}

**DvdManagementIFrame**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.Catalogue;

import Model.Dvd;

import Utility.GUIUtility;

import java.util.ArrayList;

import java.util.List;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableModel;

/\*\*

\*

\* @author Rom

\*/

public class DvdManagementIFrame extends javax.swing.JInternalFrame {

List<Catalogue> listCatalogue = new ArrayList();

Dvd selectedDvd = new Dvd();

/\*\*

\* Creates new form DvdManagementIFrame

\*/

public DvdManagementIFrame() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel2 = new javax.swing.JPanel();

jScrollPane1 = new javax.swing.JScrollPane();

dvdTable = new javax.swing.JTable();

dvdTextField = new javax.swing.JTextField();

searchButton = new javax.swing.JButton();

dvdPanel = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

titleTextField = new javax.swing.JTextField();

jLabel2 = new javax.swing.JLabel();

descriptionTextField = new javax.swing.JTextField();

jLabel3 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

languageTextField = new javax.swing.JTextField();

jLabel5 = new javax.swing.JLabel();

distributorTextField = new javax.swing.JTextField();

jLabel6 = new javax.swing.JLabel();

castTextField = new javax.swing.JTextField();

jLabel7 = new javax.swing.JLabel();

creditsTextField = new javax.swing.JTextField();

jLabel8 = new javax.swing.JLabel();

jLabel9 = new javax.swing.JLabel();

noteTextField = new javax.swing.JTextField();

jPanel3 = new javax.swing.JPanel();

addButton = new javax.swing.JButton();

updateButton = new javax.swing.JButton();

deleteButton = new javax.swing.JButton();

statusComboBox = new javax.swing.JComboBox<>();

releaseDatePicker = new org.jdesktop.swingx.JXDatePicker();

setClosable(true);

setTitle("Dvd Management");

dvdTable.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"Title", "Distributor", "Description", "Cast"

}

) {

Class[] types = new Class [] {

java.lang.String.class, java.lang.Object.class, java.lang.Object.class, java.lang.Object.class

};

public Class getColumnClass(int columnIndex) {

return types [columnIndex];

}

});

dvdTable.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

dvdTableMouseClicked(evt);

}

});

jScrollPane1.setViewportView(dvdTable);

searchButton.setText("Search");

searchButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

searchButtonActionPerformed(evt);

}

});

dvdPanel.setBorder(javax.swing.BorderFactory.createTitledBorder(null, "Dvd Details", javax.swing.border.TitledBorder.DEFAULT\_JUSTIFICATION, javax.swing.border.TitledBorder.DEFAULT\_POSITION, new java.awt.Font("Dialog", 1, 18))); // NOI18N

dvdPanel.setName(""); // NOI18N

dvdPanel.setPreferredSize(new java.awt.Dimension(520, 540));

jLabel1.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel1.setText("Title");

jLabel2.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel2.setText("Description");

jLabel3.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel3.setText("Release Date");

jLabel4.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel4.setText("Language");

jLabel5.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel5.setText("Distributor");

jLabel6.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel6.setText("Cast");

jLabel7.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel7.setText("Credits");

jLabel8.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel8.setText("Status");

jLabel9.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel9.setText("Note");

jPanel3.setMinimumSize(new java.awt.Dimension(160, 80));

jPanel3.setPreferredSize(new java.awt.Dimension(264, 80));

jPanel3.setLayout(new java.awt.GridLayout(2, 2));

addButton.setText("Add New");

addButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

addButtonActionPerformed(evt);

}

});

jPanel3.add(addButton);

updateButton.setText("Update");

updateButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

updateButtonActionPerformed(evt);

}

});

jPanel3.add(updateButton);

deleteButton.setText("Delete");

deleteButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

deleteButtonActionPerformed(evt);

}

});

jPanel3.add(deleteButton);

statusComboBox.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Available", "Borrowed", "Under Maintenance", "Not in Stock" }));

javax.swing.GroupLayout dvdPanelLayout = new javax.swing.GroupLayout(dvdPanel);

dvdPanel.setLayout(dvdPanelLayout);

dvdPanelLayout.setHorizontalGroup(

dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, dvdPanelLayout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel3, javax.swing.GroupLayout.DEFAULT\_SIZE, 378, Short.MAX\_VALUE)

.addGroup(dvdPanelLayout.createSequentialGroup()

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel8)

.addComponent(jLabel9))

.addGap(67, 67, 67)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(noteTextField)

.addComponent(statusComboBox, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)))

.addGroup(dvdPanelLayout.createSequentialGroup()

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel2)

.addComponent(jLabel1)

.addComponent(jLabel3))

.addGap(18, 18, 18)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(titleTextField, javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(descriptionTextField, javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(releaseDatePicker, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)))

.addGroup(dvdPanelLayout.createSequentialGroup()

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel4)

.addComponent(jLabel5)

.addComponent(jLabel6)

.addComponent(jLabel7))

.addGap(36, 36, 36)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(castTextField)

.addComponent(languageTextField)

.addComponent(distributorTextField)

.addComponent(creditsTextField))))

.addGap(20, 20, 20))

);

dvdPanelLayout.setVerticalGroup(

dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(dvdPanelLayout.createSequentialGroup()

.addGap(30, 30, 30)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(titleTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2)

.addComponent(descriptionTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel3)

.addComponent(releaseDatePicker, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel8)

.addComponent(statusComboBox, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(noteTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel9))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel4)

.addComponent(languageTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel5)

.addComponent(distributorTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(7, 7, 7)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(castTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel6))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(dvdPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(creditsTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel7))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 102, Short.MAX\_VALUE)

.addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 80, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(dvdTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 361, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(searchButton))

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 686, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addComponent(dvdPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 430, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(20, 20, 20))

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addContainerGap()

.addComponent(dvdPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 500, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(javax.swing.GroupLayout.Alignment.LEADING, jPanel2Layout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(dvdTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(searchButton))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 442, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGap(46, 46, 46))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

pack();

}// </editor-fold>

private void dvdTableMouseClicked(java.awt.event.MouseEvent evt) {

try {

//Get the current selected table row

int i = dvdTable.getSelectedRow();

//Get the selected book item

selectedDvd = (Dvd) listCatalogue.get(i);

//Add data to the panel controls

setDataToComponent(selectedDvd);

//Set control availability

updateButton.setEnabled(true);

deleteButton.setEnabled(true);

addButton.setEnabled(false);

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void searchButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Get the model of the table

DefaultTableModel model = (DefaultTableModel) dvdTable.getModel();

//Clean the table content

model.setRowCount(0);

//Initialize rows

Object[] rows = new Object[4];

//Initialize object

Catalogue catalogue = new Dvd();

//Search the requested book

listCatalogue = dvdTextField.getText().equals("") ? catalogue.Search() : catalogue.SearchByTitle(dvdTextField.getText());

//Add book data to the columns

for (Catalogue c : listCatalogue) {

Dvd d = (Dvd) c;

rows[0] = d.getTitle();

rows[1] = d.getDistributor();

rows[2] = d.getDescription();

rows[3] = d.getCast();

model.addRow(rows);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void addButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Check if input data is correct

String msg = GUIUtility.validateInput(dvdPanel);

if (msg.length() == 0) {

//Load book object with the panel data

Catalogue catalogue = getComponentData(new Dvd());

//Insert book into the database

catalogue.Add();

//Displays message

JOptionPane.showMessageDialog(this, "Dvd registered successfully");

//Clean data in controls

GUIUtility.cleanComponents(dvdPanel);

} else {

JOptionPane.showMessageDialog(this, msg);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void updateButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(dvdPanel);

if (msg.length() == 0) {

//Get the selected row in the table

int i = dvdTable.getSelectedRow();

//Get the title of the selected item

String title = dvdTable.getModel().getValueAt(i, 0).toString();

//Filter book based on title

Dvd dvd = (Dvd) listCatalogue.stream().filter(p -> p.getTitle().equals(title)).findFirst().get();

//Get components data

Catalogue catalogue = getComponentData(dvd);

//Update book data

catalogue.Update();

//Display message

JOptionPane.showMessageDialog(this, "Dvd updated successfully");

//Clean data in controls

GUIUtility.cleanComponents(dvdPanel);

//Enable buttons

addButton.setEnabled(true);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void deleteButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(dvdPanel);

if (msg.length() == 0) {

//Get the selected row in the table

int i = dvdTable.getSelectedRow();

//Get the current table data

TableModel model = dvdTable.getModel();

//Get the title of the selected item

String title = model.getValueAt(i, 0).toString();

//Initialize objects

Catalogue catalogue = new Dvd();

//Filter book based on title

int id = ((Dvd) (listCatalogue.stream().filter(p -> p.getTitle().equals(title)).findFirst().get())).getId();

//Update book data

catalogue.Delete(id);

//Display message

JOptionPane.showMessageDialog(this, "Dvd deleted successfully");

//Clean data in controls

GUIUtility.cleanComponents(dvdPanel);

//Enable buttons

addButton.setEnabled(true);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

/\*\*

\* Get data from the components of the panel

\* @param dvd object to be filled

\* @return filled dvd object

\* @throws Exception

\*/

private Dvd getComponentData(Dvd dvd) throws Exception {

dvd.setTitle(titleTextField.getText());

dvd.setDescription(descriptionTextField.getText());

dvd.setReleaseDate(GUIUtility.convertDateToString(releaseDatePicker.getDate()));

dvd.setStatus(statusComboBox.getSelectedItem().toString());

dvd.setNote(noteTextField.getText());

dvd.setLanguage(languageTextField.getText());

dvd.setDistributor(distributorTextField.getText());

dvd.setCast(castTextField.getText());

dvd.setCredits(creditsTextField.getText());

return dvd;

}

/\*\*

\* Set data to the components of the panel

\* @param dvd object that receives the data

\* @throws Exception

\*/

private void setDataToComponent(Dvd dvd) throws Exception {

titleTextField.setText(dvd.getTitle());

descriptionTextField.setText(dvd.getDescription());

releaseDatePicker.setDate(GUIUtility.convertToDate(dvd.getReleaseDate()));

noteTextField.setText(dvd.getNote());

languageTextField.setText(dvd.getLanguage());

distributorTextField.setText(dvd.getDistributor());

castTextField.setText(dvd.getCast());

creditsTextField.setText(dvd.getCredits());

}

// Variables declaration - do not modify

private javax.swing.JButton addButton;

private javax.swing.JTextField castTextField;

private javax.swing.JTextField creditsTextField;

private javax.swing.JButton deleteButton;

private javax.swing.JTextField descriptionTextField;

private javax.swing.JTextField distributorTextField;

private javax.swing.JPanel dvdPanel;

private javax.swing.JTable dvdTable;

private javax.swing.JTextField dvdTextField;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JLabel jLabel8;

private javax.swing.JLabel jLabel9;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTextField languageTextField;

private javax.swing.JTextField noteTextField;

private org.jdesktop.swingx.JXDatePicker releaseDatePicker;

private javax.swing.JButton searchButton;

private javax.swing.JComboBox<String> statusComboBox;

private javax.swing.JTextField titleTextField;

private javax.swing.JButton updateButton;

// End of variables declaration

}

**DvdSearchFrame**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.Catalogue;

import Model.Dvd;

import java.util.ArrayList;

import java.util.List;

import javax.swing.JInternalFrame;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

/\*\*

\*

\* @author massami

\*/

public class DvdSearchFrame extends javax.swing.JFrame {

List<Catalogue> listCatalogue = new ArrayList();

Dvd selectedDvd = new Dvd();

BorrowManagementIFrame frame;

/\*\*

\* Creates new form DvdSearchFrame

\*/

public DvdSearchFrame() {

initComponents();

}

public DvdSearchFrame(JInternalFrame internal){

initComponents();

frame = (BorrowManagementIFrame)internal;

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

dvdTextField = new javax.swing.JTextField();

searchButton = new javax.swing.JButton();

jScrollPane1 = new javax.swing.JScrollPane();

dvdTable = new javax.swing.JTable();

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE\_ON\_CLOSE);

jLabel1.setText("Search: ");

searchButton.setText("Search");

searchButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

searchButtonActionPerformed(evt);

}

});

dvdTable.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"Title", "Distributor", "Description", "Cast"

}

) {

Class[] types = new Class [] {

java.lang.String.class, java.lang.Object.class, java.lang.Object.class, java.lang.Object.class

};

public Class getColumnClass(int columnIndex) {

return types [columnIndex];

}

});

dvdTable.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

dvdTableMouseClicked(evt);

}

});

jScrollPane1.setViewportView(dvdTable);

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(dvdTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 365, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(searchButton, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 571, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE)))

.addContainerGap())

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(dvdTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(searchButton))

.addGap(18, 18, 18)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 430, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addContainerGap())

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void searchButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Get the model of the table

DefaultTableModel model = (DefaultTableModel) dvdTable.getModel();

//Clean the table content

model.setRowCount(0);

//Initialize rows

Object[] rows = new Object[4];

//Initialize Book object

Catalogue catalogue = new Dvd();

//Search the requested book

listCatalogue = dvdTextField.getText().equals("") ? catalogue.SearchAvailableItems(): catalogue.SearchAvailableItemsByTitle(dvdTextField.getText());

//Add book data to the table

for (Catalogue c : listCatalogue) {

Dvd d = (Dvd) c;

rows[0] = d.getTitle();

rows[1] = d.getDistributor();

rows[2] = d.getDescription();

rows[3] = d.getCast();

model.addRow(rows);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void dvdTableMouseClicked(java.awt.event.MouseEvent evt) {

try {

//Get the current selected table row

int i = dvdTable.getSelectedRow();

//Get the selected book item

selectedDvd = (Dvd) listCatalogue.get(i);

//Add selected book to table

frame.addItem(selectedDvd);

//Close Book search window

this.dispose();

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(DvdSearchFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(DvdSearchFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(DvdSearchFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(DvdSearchFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new DvdSearchFrame().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JTable dvdTable;

private javax.swing.JTextField dvdTextField;

private javax.swing.JLabel jLabel1;

private javax.swing.JPanel jPanel1;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JButton searchButton;

// End of variables declaration

}

**Login**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.Person;

import Model.User;

import Shared.SharedData;

import java.util.ArrayList;

import javax.swing.JOptionPane;

/\*\*

\*

\* @author massami

\*/

public class Login extends javax.swing.JFrame {

/\*\*

\* Creates new form Login

\*/

public Login() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jButton1 = new javax.swing.JButton();

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

usernameTextField = new javax.swing.JTextField();

passwordTextField = new javax.swing.JTextField();

jLabel2 = new javax.swing.JLabel();

loginButton = new javax.swing.JButton();

jButton1.setText("jButton1");

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jLabel1.setText("Username:");

jLabel2.setText("Password:");

loginButton.setText("Login");

loginButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

loginButtonActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(loginButton, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel1)

.addComponent(jLabel2))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(passwordTextField, javax.swing.GroupLayout.DEFAULT\_SIZE, 178, Short.MAX\_VALUE)

.addComponent(usernameTextField))))

.addContainerGap())

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(usernameTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2)

.addComponent(passwordTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(50, 50, 50)

.addComponent(loginButton)

.addContainerGap(22, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addContainerGap())

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addContainerGap())

);

pack();

}// </editor-fold>

private void loginButtonActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try {

//Get components data

User user = new User();

//Get the user data

String username = usernameTextField.getText();

String password = passwordTextField.getText();

//Update book data

ArrayList<Person> p = user.CheckPassword(username,password);

if(p.isEmpty()){

JOptionPane.showMessageDialog(this, "Password is not correct");

}

else {

//Save user data

SharedData.username = username;

//Close login window

this.dispose();

//Start application

new ApplicationGUI().setVisible(true);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, "Password is not correct");

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new Login().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JPanel jPanel1;

private javax.swing.JButton loginButton;

private javax.swing.JTextField passwordTextField;

private javax.swing.JTextField usernameTextField;

// End of variables declaration

}

**MemberManagementIFrame**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.Member;

import Model.Person;

import Utility.GUIUtility;

import java.util.ArrayList;

import java.util.List;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableModel;

/\*\*

\*

\* @author Rom

\*/

public class MemberManagementIFrame extends javax.swing.JInternalFrame {

List<Person> listPerson = new ArrayList();

Member selectedMember = new Member();

/\*\*

\* Creates new form MemberManagementInternalFrame

\*/

public MemberManagementIFrame() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel2 = new javax.swing.JPanel();

jScrollPane1 = new javax.swing.JScrollPane();

memberTable = new javax.swing.JTable();

searchTextField = new javax.swing.JTextField();

searchButton = new javax.swing.JButton();

memberPanel = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

firstNameTextField = new javax.swing.JTextField();

jLabel2 = new javax.swing.JLabel();

lastNameTextField = new javax.swing.JTextField();

jLabel3 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

addressTextField = new javax.swing.JTextField();

jLabel5 = new javax.swing.JLabel();

suburbTextField = new javax.swing.JTextField();

jLabel6 = new javax.swing.JLabel();

cityTextField = new javax.swing.JTextField();

jLabel7 = new javax.swing.JLabel();

postalCodeTextField = new javax.swing.JTextField();

jLabel8 = new javax.swing.JLabel();

emailAddressTextField = new javax.swing.JTextField();

jLabel9 = new javax.swing.JLabel();

contactNumberTextField = new javax.swing.JTextField();

jPanel3 = new javax.swing.JPanel();

addButton = new javax.swing.JButton();

updateButton = new javax.swing.JButton();

activationButton = new javax.swing.JButton();

transactionHistoryButton = new javax.swing.JButton();

birthDatePicker = new org.jdesktop.swingx.JXDatePicker();

setClosable(true);

setTitle("Member Management");

setMaximumSize(new java.awt.Dimension(1180, 580));

setMinimumSize(new java.awt.Dimension(1180, 580));

setNormalBounds(new java.awt.Rectangle(0, 0, 80, 0));

setPreferredSize(new java.awt.Dimension(1180, 580));

memberTable.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"First Name", "Last Name", "Email"

}

) {

Class[] types = new Class [] {

java.lang.String.class, java.lang.Object.class, java.lang.Object.class

};

public Class getColumnClass(int columnIndex) {

return types [columnIndex];

}

});

memberTable.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

memberTableMouseClicked(evt);

}

});

jScrollPane1.setViewportView(memberTable);

searchButton.setText("Search");

searchButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

searchButtonActionPerformed(evt);

}

});

memberPanel.setBorder(javax.swing.BorderFactory.createTitledBorder(null, "Member Details", javax.swing.border.TitledBorder.DEFAULT\_JUSTIFICATION, javax.swing.border.TitledBorder.DEFAULT\_POSITION, new java.awt.Font("Dialog", 1, 18))); // NOI18N

memberPanel.setName(""); // NOI18N

memberPanel.setPreferredSize(new java.awt.Dimension(520, 540));

jLabel1.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel1.setText("First Name");

firstNameTextField.setName("firstName"); // NOI18N

jLabel2.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel2.setText("Last Name");

lastNameTextField.setName("lastName"); // NOI18N

jLabel3.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel3.setText("Birthdate");

jLabel4.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel4.setText("Address");

addressTextField.setName("address"); // NOI18N

jLabel5.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel5.setText("Suburb");

suburbTextField.setName("suburb"); // NOI18N

jLabel6.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel6.setText("City");

cityTextField.setName("city"); // NOI18N

jLabel7.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel7.setText("Postal Code");

postalCodeTextField.setName("postalCode"); // NOI18N

jLabel8.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel8.setText("Email Address");

emailAddressTextField.setName("emailAddress"); // NOI18N

jLabel9.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel9.setText("Contact Number");

contactNumberTextField.setName("contactNumber"); // NOI18N

jPanel3.setLayout(new java.awt.GridLayout(2, 2));

addButton.setText("Add New");

addButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

addButtonActionPerformed(evt);

}

});

jPanel3.add(addButton);

updateButton.setText("Update");

updateButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

updateButtonActionPerformed(evt);

}

});

jPanel3.add(updateButton);

activationButton.setBackground(new java.awt.Color(204, 0, 0));

activationButton.setForeground(new java.awt.Color(255, 255, 255));

activationButton.setText("Deactivate");

activationButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

activationButtonActionPerformed(evt);

}

});

jPanel3.add(activationButton);

transactionHistoryButton.setText("Transaction History");

transactionHistoryButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

transactionHistoryButtonActionPerformed(evt);

}

});

jPanel3.add(transactionHistoryButton);

birthDatePicker.setName("birthDate"); // NOI18N

javax.swing.GroupLayout memberPanelLayout = new javax.swing.GroupLayout(memberPanel);

memberPanel.setLayout(memberPanelLayout);

memberPanelLayout.setHorizontalGroup(

memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, memberPanelLayout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(memberPanelLayout.createSequentialGroup()

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel4)

.addComponent(jLabel5))

.addGap(73, 73, 73)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(addressTextField, javax.swing.GroupLayout.DEFAULT\_SIZE, 352, Short.MAX\_VALUE)

.addComponent(suburbTextField)))

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jPanel3, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, 483, Short.MAX\_VALUE)

.addGroup(memberPanelLayout.createSequentialGroup()

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel1)

.addComponent(jLabel2)

.addComponent(jLabel3))

.addGap(57, 57, 57)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(lastNameTextField, javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(firstNameTextField)

.addComponent(birthDatePicker, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)))

.addGroup(memberPanelLayout.createSequentialGroup()

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel7)

.addComponent(jLabel6))

.addGap(47, 47, 47)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(cityTextField)

.addComponent(postalCodeTextField))))

.addGroup(memberPanelLayout.createSequentialGroup()

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel9)

.addComponent(jLabel8))

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(memberPanelLayout.createSequentialGroup()

.addGap(18, 18, 18)

.addComponent(contactNumberTextField))

.addGroup(memberPanelLayout.createSequentialGroup()

.addGap(18, 18, 18)

.addComponent(emailAddressTextField)))))

.addContainerGap())

);

memberPanelLayout.setVerticalGroup(

memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(memberPanelLayout.createSequentialGroup()

.addGap(30, 30, 30)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(firstNameTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2)

.addComponent(lastNameTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel3)

.addComponent(birthDatePicker, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(emailAddressTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel8))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(contactNumberTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel9))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(addressTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel4))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel5)

.addComponent(suburbTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel6)

.addComponent(cityTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(memberPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(postalCodeTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel7))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 110, Short.MAX\_VALUE)

.addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 80, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(searchTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 361, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(searchButton, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 501, Short.MAX\_VALUE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(memberPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 525, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(memberPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 500, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(searchTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(searchButton))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 457, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addContainerGap(37, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

pack();

}// </editor-fold>

private void addButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Check if input data is correct

String msg = GUIUtility.validateInput(memberPanel);

if (msg.length() == 0) {

//Load book object with the panel data

Person person = getComponentData(new Member());

//Insert book into the database

person.Add();

//Displays message

JOptionPane.showMessageDialog(this, "Member registered successfully");

//Clean data in controls

GUIUtility.cleanComponents(memberPanel);

//Reset the table`s content

GUIUtility.cleanTable(memberTable);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void searchButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Get the model of the table

DefaultTableModel model = (DefaultTableModel) memberTable.getModel();

//Clean the table content

model.setRowCount(0);

//Initialize rows

Object[] rows = new Object[3];

//Initialize Book object

Person person = new Member();

//Search the requested book

listPerson = searchTextField.getText().equals("") ? person.Search() : person.SearchByName(searchTextField.getText());

//Add book data to the table

for (Person p : listPerson) {

Member m = (Member) p;

rows[0] = m.getFirstName();

rows[1] = m.getLastName();

rows[2] = m.getEmailAddress();

model.addRow(rows);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void memberTableMouseClicked(java.awt.event.MouseEvent evt) {

try {

//Get the current selected table row

int i = memberTable.getSelectedRow();

//Get the selected member item

selectedMember = (Member) listPerson.get(i);

//Add data to the panel controls

setDataToComponent(selectedMember);

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void updateButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(memberPanel);

if (msg.length() == 0) {

//Get the selected row in the table

int i = memberTable.getSelectedRow();

//Get the name of the selected item

String firstName = memberTable.getModel().getValueAt(i, 0).toString();

String lastName = memberTable.getModel().getValueAt(i, 1).toString();

//Get the selected book

Member member = (Member) listPerson.stream().filter(p -> p.getFirstName().equals(firstName) && p.getLastName().equals(lastName)).findFirst().get();

//Get components data

Person person = getComponentData(member);

//Update book data

person.Update();

//Display message

JOptionPane.showMessageDialog(this, "Member updated successfully");

//Clean data in controls

GUIUtility.cleanComponents(memberPanel);

//Reset the table`s content

GUIUtility.cleanTable(memberTable);

//Enable buttons

addButton.setEnabled(true);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void activationButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(memberPanel);

if (msg.length() == 0) {

//Get the selected row in the table

int i = memberTable.getSelectedRow();

//Get the current table data

TableModel model = memberTable.getModel();

//Get the title of the selected item

String firstName = model.getValueAt(i, 0).toString();

String lastName = model.getValueAt(i, 1).toString();

//Initialize objects

Person person = new Member();

//Filter book based on title

int id = ((Member) (listPerson.stream().filter(p -> p.getFirstName().equals(firstName)&&p.getLastName().equals(lastName)).findFirst().get())).getId();

//Update book data

person.Deactivate(id);

//Display message

JOptionPane.showMessageDialog(this, "Member deactivated successfully");

//Clean data in controls

GUIUtility.cleanComponents(memberPanel);

//Enable buttons

addButton.setEnabled(true);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void transactionHistoryButtonActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

TransactionHistory history = new TransactionHistory();

history.setVisible(true);

}

/\*\*

\* Get the data from the components

\*

\* @param member object to be filled

\* @return filled object

\* @throws Exception

\*/

private Member getComponentData(Member member) throws Exception {

member.setFirstName(firstNameTextField.getText());

member.setLastName(lastNameTextField.getText());

member.setBirthdate(GUIUtility.convertDateToString(birthDatePicker.getDate()));

member.setEmailAddress(emailAddressTextField.getText());

member.setContactNumber(contactNumberTextField.getText());

member.setAddress(addressTextField.getText());

member.setSuburb(suburbTextField.getText());

member.setCity(cityTextField.getText());

member.setPostalCode(postalCodeTextField.getText());

return member;

}

/\*\*

\* Set data from the object to the components of the GUI

\*

\* @param member get data from

\* @throws Exception

\*/

private void setDataToComponent(Member member) throws Exception {

firstNameTextField.setText(member.getFirstName());

lastNameTextField.setText(member.getLastName());

birthDatePicker.setDate(GUIUtility.convertToDate(member.getBirthdate()));

emailAddressTextField.setText(member.getEmailAddress());

contactNumberTextField.setText(member.getContactNumber());

addressTextField.setText(member.getAddress());

suburbTextField.setText(member.getSuburb());

cityTextField.setText(member.getCity());

postalCodeTextField.setText(member.getPostalCode());

}

// Variables declaration - do not modify

private javax.swing.JButton activationButton;

private javax.swing.JButton addButton;

private javax.swing.JTextField addressTextField;

private org.jdesktop.swingx.JXDatePicker birthDatePicker;

private javax.swing.JTextField cityTextField;

private javax.swing.JTextField contactNumberTextField;

private javax.swing.JTextField emailAddressTextField;

private javax.swing.JTextField firstNameTextField;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JLabel jLabel8;

private javax.swing.JLabel jLabel9;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTextField lastNameTextField;

private javax.swing.JPanel memberPanel;

private javax.swing.JTable memberTable;

private javax.swing.JTextField postalCodeTextField;

private javax.swing.JButton searchButton;

private javax.swing.JTextField searchTextField;

private javax.swing.JTextField suburbTextField;

private javax.swing.JButton transactionHistoryButton;

private javax.swing.JButton updateButton;

// End of variables declaration

}

**TransactionHistory**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.History;

import Model.Member;

import java.util.ArrayList;

import javax.swing.DefaultComboBoxModel;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

/\*\*

\*

\* @author massami

\*/

public class TransactionHistory extends javax.swing.JFrame {

ArrayList<String> memberList = new ArrayList();

ArrayList<History> listHistory = new ArrayList();

/\*\*

\* Creates new form TransactionHistory

\*/

public TransactionHistory() {

initComponents();

LoadComboBox();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

memberComboBox = new javax.swing.JComboBox<>();

searchButton = new javax.swing.JButton();

jScrollPane1 = new javax.swing.JScrollPane();

transactionTable = new javax.swing.JTable();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jLabel1.setText("Search member:");

memberComboBox.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

searchButton.setText("Search");

searchButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

searchButtonActionPerformed(evt);

}

});

transactionTable.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"Title", "Due Date", "Borrow Date"

}

) {

boolean[] canEdit = new boolean [] {

false, false, false

};

public boolean isCellEditable(int rowIndex, int columnIndex) {

return canEdit [columnIndex];

}

});

jScrollPane1.setViewportView(transactionTable);

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 623, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(memberComboBox, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(searchButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 121, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(memberComboBox, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(searchButton))

.addGap(18, 18, 18)

.addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 322, Short.MAX\_VALUE)

.addContainerGap())

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGap(169, 169, 169))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void searchButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Get the model of the table

DefaultTableModel model = (DefaultTableModel) transactionTable.getModel();

//Clean the table content

model.setRowCount(0);

//Initialize rows

Object[] rows = new Object[4];

//Initialize history object

History history = new History();

//Search the requested book

listHistory = history.searchTransactionHistory(memberComboBox.getSelectedItem().toString());

//Add book data to the table

for (History h : listHistory) {

rows[0] = h.getTitle();

rows[1] = h.getDueDate();

rows[2] = h.getBorrowDate();

model.addRow(rows);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this,e.getMessage());

}

}

/\*\*

\* Load members data to the combobox

\*/

private void LoadComboBox() {

try {

Member member = new Member();

memberList = member.SearchMemberName();

memberComboBox.setModel((new DefaultComboBoxModel(memberList.toArray())));

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(TransactionHistory.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(TransactionHistory.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(TransactionHistory.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(TransactionHistory.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new TransactionHistory().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JLabel jLabel1;

private javax.swing.JPanel jPanel1;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JComboBox<String> memberComboBox;

private javax.swing.JButton searchButton;

private javax.swing.JTable transactionTable;

// End of variables declaration

}

**UserManagementIFrame**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package GUI;

import Model.Person;

import Model.User;

import Utility.GUIUtility;

import java.util.ArrayList;

import java.util.List;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableModel;

/\*\*

\*

\* @author Rom

\*/

public class UserManagementIFrame extends javax.swing.JInternalFrame {

List<Person> listPerson = new ArrayList();

User selectedUser = new User();

/\*\*

\* Creates new form UserManagementIFrame

\*/

public UserManagementIFrame() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel2 = new javax.swing.JPanel();

jScrollPane1 = new javax.swing.JScrollPane();

userTable = new javax.swing.JTable();

searchTextField = new javax.swing.JTextField();

searchButton = new javax.swing.JButton();

userPanel = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

usernameTextField = new javax.swing.JTextField();

jLabel2 = new javax.swing.JLabel();

firstNameTextField = new javax.swing.JTextField();

jLabel8 = new javax.swing.JLabel();

emailAddressTextField = new javax.swing.JTextField();

jLabel9 = new javax.swing.JLabel();

contactNumberTextField = new javax.swing.JTextField();

lastNameTextField = new javax.swing.JTextField();

jLabel10 = new javax.swing.JLabel();

jPanel3 = new javax.swing.JPanel();

addButton = new javax.swing.JButton();

updateButton = new javax.swing.JButton();

activationButton = new javax.swing.JButton();

resetPasswordButton = new javax.swing.JButton();

setClosable(true);

setTitle("User Management");

setMaximumSize(new java.awt.Dimension(1176, 454));

setMinimumSize(new java.awt.Dimension(1176, 454));

setPreferredSize(new java.awt.Dimension(1176, 454));

userTable.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"First Name", "Last Name", "Email"

}

) {

Class[] types = new Class [] {

java.lang.String.class, java.lang.Object.class, java.lang.Object.class

};

public Class getColumnClass(int columnIndex) {

return types [columnIndex];

}

});

userTable.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

userTableMouseClicked(evt);

}

});

jScrollPane1.setViewportView(userTable);

searchButton.setText("Search");

searchButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

searchButtonActionPerformed(evt);

}

});

userPanel.setBorder(javax.swing.BorderFactory.createTitledBorder(null, "User Details", javax.swing.border.TitledBorder.DEFAULT\_JUSTIFICATION, javax.swing.border.TitledBorder.DEFAULT\_POSITION, new java.awt.Font("Dialog", 1, 18))); // NOI18N

userPanel.setName(""); // NOI18N

userPanel.setPreferredSize(new java.awt.Dimension(520, 540));

jLabel1.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel1.setText("Username");

usernameTextField.setName("username"); // NOI18N

jLabel2.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel2.setText("First Name");

firstNameTextField.setName("firstName"); // NOI18N

jLabel8.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel8.setText("Email Address");

emailAddressTextField.setName("emailAddress"); // NOI18N

jLabel9.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel9.setText("Contact Number");

contactNumberTextField.setName("contactNumber"); // NOI18N

lastNameTextField.setName("lastName"); // NOI18N

jLabel10.setFont(new java.awt.Font("Dialog", 1, 14)); // NOI18N

jLabel10.setText("Last Name");

jPanel3.setLayout(new java.awt.GridLayout(2, 2));

addButton.setText("Add New");

addButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

addButtonActionPerformed(evt);

}

});

jPanel3.add(addButton);

updateButton.setText("Update");

updateButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

updateButtonActionPerformed(evt);

}

});

jPanel3.add(updateButton);

activationButton.setBackground(new java.awt.Color(204, 0, 0));

activationButton.setForeground(new java.awt.Color(255, 255, 255));

activationButton.setText("Deactivate");

activationButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

activationButtonActionPerformed(evt);

}

});

jPanel3.add(activationButton);

resetPasswordButton.setText("Reset Password");

resetPasswordButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

resetPasswordButtonActionPerformed(evt);

}

});

jPanel3.add(resetPasswordButton);

javax.swing.GroupLayout userPanelLayout = new javax.swing.GroupLayout(userPanel);

userPanel.setLayout(userPanelLayout);

userPanelLayout.setHorizontalGroup(

userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(userPanelLayout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel3, javax.swing.GroupLayout.DEFAULT\_SIZE, 378, Short.MAX\_VALUE)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, userPanelLayout.createSequentialGroup()

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel10)

.addComponent(jLabel2)

.addComponent(jLabel1))

.addGap(54, 54, 54)

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(lastNameTextField)

.addComponent(firstNameTextField)

.addComponent(usernameTextField)))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, userPanelLayout.createSequentialGroup()

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel9)

.addComponent(jLabel8))

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(userPanelLayout.createSequentialGroup()

.addGap(18, 18, 18)

.addComponent(contactNumberTextField))

.addGroup(userPanelLayout.createSequentialGroup()

.addGap(16, 16, 16)

.addComponent(emailAddressTextField)))))

.addGap(20, 20, 20))

);

userPanelLayout.setVerticalGroup(

userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(userPanelLayout.createSequentialGroup()

.addGap(30, 30, 30)

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(usernameTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2)

.addComponent(firstNameTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel10)

.addComponent(lastNameTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel8)

.addComponent(emailAddressTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(5, 5, 5)

.addGroup(userPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(contactNumberTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel9))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 98, Short.MAX\_VALUE)

.addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 80, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(20, 20, 20)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addComponent(searchTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 361, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(searchButton))

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 686, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addComponent(userPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 430, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(20, Short.MAX\_VALUE))

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(8, 8, 8)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(searchTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(searchButton))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 0, Short.MAX\_VALUE))

.addGroup(jPanel2Layout.createSequentialGroup()

.addContainerGap()

.addComponent(userPanel, javax.swing.GroupLayout.PREFERRED\_SIZE, 380, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGap(46, 46, 46))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

pack();

}// </editor-fold>

private void userTableMouseClicked(java.awt.event.MouseEvent evt) {

try {

//Get the current selected table row

int i = userTable.getSelectedRow();

//Get the selected member item

selectedUser = (User) listPerson.get(i);

//Add data to the panel controls

setDataToComponent(selectedUser);

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void searchButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Get the model of the table

DefaultTableModel model = (DefaultTableModel) userTable.getModel();

//Clean the table content

model.setRowCount(0);

//Initialize rows

Object[] rows = new Object[3];

//Initialize Book object

Person person = new User();

//Search the requested book

listPerson = searchTextField.getText().equals("") ? person.Search() : person.SearchByName(searchTextField.getText());

//Add book data to the table

for (Person p : listPerson) {

User u = (User) p;

rows[0] = u.getFirstName();

rows[1] = u.getLastName();

rows[2] = u.getEmailAddress();

model.addRow(rows);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void addButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Check if input data is correct

String msg = GUIUtility.validateInput(userPanel);

if (msg.length() == 0) {

//Load book object with the panel data

Person person = getComponentData(new User());

//Insert book into the database

person.Add();

//Displays message

JOptionPane.showMessageDialog(this, "User registered successfully");

//Clean data in controls

GUIUtility.cleanComponents(userPanel);

//Reset the table`s content

GUIUtility.cleanTable(userTable);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void activationButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(userPanel);

if (msg.length() == 0) {

//Get the selected row in the table

int i = userTable.getSelectedRow();

//Get the current table data

TableModel model = userTable.getModel();

//Get the title of the selected item

String firstName = model.getValueAt(i, 0).toString();

String lastName = model.getValueAt(i, 1).toString();

//Initialize objects

Person person = new User();

//Filter book based on title

int id = ((User) (listPerson.stream().filter(p -> p.getFirstName().equals(firstName) && p.getLastName().equals(lastName)).findFirst().get())).getId();

//Update book data

person.Deactivate(id);

//Display message

JOptionPane.showMessageDialog(this, "User deactivated successfully");

//Clean data in controls

GUIUtility.cleanComponents(userPanel);

//Enable buttons

addButton.setEnabled(true);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void updateButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(userPanel);

if (msg.length() == 0) {

//Get the selected row in the table

int i = userTable.getSelectedRow();

//Get the name of the selected item

String firstName = userTable.getModel().getValueAt(i, 0).toString();

String lastName = userTable.getModel().getValueAt(i, 1).toString();

//Get the selected book

User user = (User) listPerson.stream().filter(p -> p.getFirstName().equals(firstName) && p.getLastName().equals(lastName)).findFirst().get();

//Get components data

Person person = getComponentData(user);

//Update book data

person.Update();

//Display message

JOptionPane.showMessageDialog(this, "User updated successfully");

//Clean data in controls

GUIUtility.cleanComponents(userPanel);

//Reset the table`s content

GUIUtility.cleanTable(userTable);

//Enable buttons

addButton.setEnabled(true);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

private void resetPasswordButtonActionPerformed(java.awt.event.ActionEvent evt) {

try {

//Message with the field validation

String msg = GUIUtility.validateInput(userPanel);

if (msg.length() == 0) {

//Get the selected row in the table

int i = userTable.getSelectedRow();

//Get the name of the selected item

String firstName = userTable.getModel().getValueAt(i, 0).toString();

String lastName = userTable.getModel().getValueAt(i, 1).toString();

//Get the selected user

int id = listPerson.stream().filter(p -> p.getFirstName().equals(firstName) && p.getLastName().equals(lastName)).findFirst().get().getId();

//Instantiate new object

User user = new User();

//Update book data

user.ResetPassword(id);

//Display message

JOptionPane.showMessageDialog(this, "Password reseted successfully");

//Clean data in controls

GUIUtility.cleanComponents(userPanel);

//Reset the table`s content

GUIUtility.cleanTable(userTable);

}

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e.getMessage());

}

}

/\*\*

\* Get the data from the components

\*

\* @param member object to be filled

\* @return filled object

\* @throws Exception

\*/

private User getComponentData(User user) throws Exception {

user.setUsername(usernameTextField.getText());

user.setFirstName(firstNameTextField.getText());

user.setLastName(lastNameTextField.getText());

user.setEmailAddress(emailAddressTextField.getText());

user.setContactNumber(contactNumberTextField.getText());

user.setPassword("123"); //Default password

return user;

}

/\*\*

\* Set data from the object to the components of the GUI

\*

\* @param member get data from

\* @throws Exception

\*/

private void setDataToComponent(User user) throws Exception {

usernameTextField.setText(user.getUsername());

firstNameTextField.setText(user.getFirstName());

lastNameTextField.setText(user.getLastName());

emailAddressTextField.setText(user.getEmailAddress());

contactNumberTextField.setText(user.getContactNumber());

}

// Variables declaration - do not modify

private javax.swing.JButton activationButton;

private javax.swing.JButton addButton;

private javax.swing.JTextField contactNumberTextField;

private javax.swing.JTextField emailAddressTextField;

private javax.swing.JTextField firstNameTextField;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel10;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel8;

private javax.swing.JLabel jLabel9;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTextField lastNameTextField;

private javax.swing.JButton resetPasswordButton;

private javax.swing.JButton searchButton;

private javax.swing.JTextField searchTextField;

private javax.swing.JButton updateButton;

private javax.swing.JPanel userPanel;

private javax.swing.JTable userTable;

private javax.swing.JTextField usernameTextField;

// End of variables declaration

}

**Model Package**

**Book**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Model;

import DAL.BookDAL;

import java.util.ArrayList;

/\*\*

\*

\* @author Rom

\*/

public class Book extends Catalogue {

private int id;

private String author;

private String genre;

private String isbn;

public Book(String author, String genre, String isbn, String title, String description, String releaseDate, String status, String note, String language) {

super(title, description, releaseDate, status, note, language);

this.author = author;

this.genre = genre;

this.isbn = isbn;

}

public Book(int id, String author, String genre, String isbn, String title, String description, String releaseDate, String status, String note, String language) {

super(title, description, releaseDate, status, note, language);

this.id = id;

this.author = author;

this.genre = genre;

this.isbn = isbn;

}

public Book(int catalogueId,int bookId, String author, String genre, String isbn, String title, String description, String releaseDate, String status, String note, String language) {

super(catalogueId, title, description, releaseDate, status, note, language);

this.id = bookId;

this.author = author;

this.genre = genre;

this.isbn = isbn;

}

public Book() {

}

public int getBookId() {

return id;

}

public String getAuthor() {

return author;

}

public void setAuthor(String author) {

this.author = author;

}

public String getGenre() {

return genre;

}

public void setGenre(String genre) {

this.genre = genre;

}

public String getIsbn() {

return isbn;

}

public void setIsbn(String isbn) {

this.isbn = isbn;

}

@Override

public void Add() throws Exception {

BookDAL.addBook(this);

}

@Override

public void Update() throws Exception {

BookDAL.updateBook(this);

}

@Override

public void Delete(int id) throws Exception {

BookDAL.deleteBook(id);

}

@Override

public ArrayList<Catalogue> Search() throws Exception {

return BookDAL.getAllBook();

}

@Override

public ArrayList<Catalogue> SearchByTitle(String title) throws Exception {

return BookDAL.getBookByTitle(title);

}

@Override

public boolean equals(Object other){

if(!(other instanceof Book)){

return false;

}

Book b = (Book) other;

return this.author.equals(b.getAuthor()) &&

this.genre.equals(b.getGenre()) &&

this.isbn.equals(b.getIsbn());

}

@Override

public ArrayList<Catalogue> SearchAvailableItems() throws Exception {

return BookDAL.getAvailableBook();

}

@Override

public ArrayList<Catalogue> SearchAvailableItemsByTitle(String title) throws Exception {

return BookDAL.getAvailableBookByTitle(title);

}

}

**Borrow**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Model;

import DAL.BorrowDAL;

import java.util.ArrayList;

/\*\*

\*

\* @author massami

\*/

public class Borrow {

private int id;

private String borrowDate;

private String dueDate;

private String returnDate;

private int memberId;

private int catalogueId;

public Borrow(String borrowDate, String dueDate, String returnDate, int memberId, int catalogueId) {

this.borrowDate = borrowDate;

this.dueDate = dueDate;

this.returnDate = returnDate;

this.memberId = memberId;

this.catalogueId = catalogueId;

}

public Borrow(){

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getBorrowDate() {

return borrowDate;

}

public void setBorrowDate(String borrowDate) {

this.borrowDate = borrowDate;

}

public String getDueDate() {

return dueDate;

}

public void setDueDate(String dueDate) {

this.dueDate = dueDate;

}

public String getReturnDate() {

return returnDate;

}

public void setReturnDate(String returnDate) {

this.returnDate = returnDate;

}

public int getMemberId() {

return memberId;

}

public void setMemberId(int memberId) {

this.memberId = memberId;

}

public int getCatalogueId() {

return catalogueId;

}

public void setCatalogueId(int catalogueId) {

this.catalogueId = catalogueId;

}

public void addBorrow(Borrow borrow) throws Exception{

BorrowDAL.addBorrow(borrow);

}

public void returnItem(Borrow borrow) throws Exception{

BorrowDAL.returnItem(borrow);

}

public int searchMemberId(String name) throws Exception{

return BorrowDAL.getMemberId(name);

}

public ArrayList<Catalogue> searchBorrowedItem(String name) throws Exception{

return BorrowDAL.getBorrowedItem(name);

}

}

**Catalogue**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Model;

import java.util.ArrayList;

/\*\*

\*

\* @author Rom

\*/

public abstract class Catalogue {

private int id;

private String title;

private String description;

private String releaseDate;

private String status;

private String note;

private String language;

public Catalogue(String title, String description, String releaseDate, String status, String note, String language) {

this.title = title;

this.description = description;

this.releaseDate = releaseDate;

this.status = status;

this.note = note;

this.language = language;

}

public Catalogue(int id, String title, String description, String releaseDate, String status, String note, String language) {

this.id = id;

this.title = title;

this.description = description;

this.releaseDate = releaseDate;

this.status = status;

this.note = note;

this.language = language;

}

public Catalogue() {

}

public void setId(int id){

this.id = id;

}

public int getId(){

return id;

}

public String getTitle() {

return title;

}

public void setTitle(String title) {

this.title = title;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

public String getReleaseDate() {

return releaseDate;

}

public void setReleaseDate(String releaseDate) {

this.releaseDate = releaseDate;

}

public String getStatus() {

return status;

}

public void setStatus(String status) throws Exception {

if (status.equals("Available") || status.equals("Borrowed") || status.equals("Under Maintenance") || status.equals("Not in Stock")) {

this.status = status;

} else {

throw new Exception("Please enter correct status");

}

}

public String getNote() {

return note;

}

public void setNote(String note) {

this.note = note;

}

public String getLanguage() {

return language;

}

public void setLanguage(String language) {

this.language = language;

}

public abstract void Add() throws Exception;

public abstract void Update() throws Exception;

public abstract void Delete(int id) throws Exception;

public abstract ArrayList<Catalogue> Search() throws Exception;

public abstract ArrayList<Catalogue> SearchByTitle(String title) throws Exception;

public abstract ArrayList<Catalogue> SearchAvailableItems() throws Exception;

public abstract ArrayList<Catalogue> SearchAvailableItemsByTitle(String title) throws Exception;

@Override

public boolean equals(Object other){

if(!(other instanceof Catalogue)){

return false;

}

Catalogue cat = (Catalogue) other;

return this.title.equals(cat.getTitle()) &&

this.description.equals(cat.getDescription()) &&

this.releaseDate.equals(cat.getReleaseDate()) &&

this.status.equals(cat.getStatus()) &&

this.note.equals(cat.getNote()) &&

this.language.equals(cat.getLanguage());

}

}

**Dvd**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Model;

import DAL.DvdDAL;

import java.util.ArrayList;

/\*\*

\*

\* @author Rom

\*/

public class Dvd extends Catalogue {

private int id;

private String distributor;

private String cast;

private String credits;

private int catalogueId;

public Dvd(int id, String distributor, String cast, String credits, String title, String description, String releaseDate, String status, String note, String language) {

super(title, description, releaseDate, status, note, language);

this.id = id;

this.distributor = distributor;

this.cast = cast;

this.credits = credits;

}

public Dvd(int catalogueId, int id, String distributor, String cast, String credits, String title, String description, String releaseDate, String status, String note, String language) {

super(catalogueId, title, description, releaseDate, status, note, language);

this.id = id;

this.distributor = distributor;

this.cast = cast;

this.credits = credits;

}

public Dvd(){

}

public int getDvdId(){

return id;

}

public String getDistributor() {

return distributor;

}

public void setDistributor(String distributor) {

this.distributor = distributor;

}

public String getCast() {

return cast;

}

public void setCast(String cast) {

this.cast = cast;

}

public String getCredits() {

return credits;

}

public void setCredits(String credits) {

this.credits = credits;

}

public int getCatalogueId(){

return this.catalogueId;

}

public void setCatalogueId(int id){

this.catalogueId = id;

}

@Override

public void Add() throws Exception{

DvdDAL.addDvd(this);

}

@Override

public void Update() throws Exception{

DvdDAL.updateDvd(this);

}

@Override

public void Delete(int id) throws Exception{

DvdDAL.deleteDvd(id);

}

@Override

public ArrayList<Catalogue> Search() throws Exception {

return DvdDAL.getAllDvd();

}

@Override

public ArrayList<Catalogue> SearchByTitle(String title) throws Exception {

return DvdDAL.getDvdByTitle(title);

}

@Override

public ArrayList<Catalogue> SearchAvailableItems() throws Exception {

return DvdDAL.getAvailableDvd();

}

@Override

public ArrayList<Catalogue> SearchAvailableItemsByTitle(String title) throws Exception {

return DvdDAL.getAvailableDvdByTitle(title);

}

}

**History**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Model;

import DAL.HistoryDAL;

import java.util.ArrayList;

/\*\*

\*

\* @author massami

\*/

public class History {

private String title;

private String dueDate;

private String borrowDate;

private String name;

public History(String title, String dueDate, String borrowDate, String name) {

this.title = title;

this.dueDate = dueDate;

this.borrowDate = borrowDate;

this.name = name;

}

public History(){

}

public String getTitle() {

return title;

}

public void setTitle(String title) {

this.title = title;

}

public String getDueDate() {

return dueDate;

}

public void setDueDate(String dueDate) {

this.dueDate = dueDate;

}

public String getBorrowDate() {

return borrowDate;

}

public void setBorrowDate(String borrowDate) {

this.borrowDate = borrowDate;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public ArrayList<History> searchTransactionHistory(String name) throws Exception{

return HistoryDAL.getTransactionHistory(name);

}

}

**Member**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Model;

import DAL.MemberDAL;

import java.util.ArrayList;

/\*\*

\*

\* @author Rom

\*/

public class Member extends Person {

private String birthdate;

public Member(int id, String firstName, String lastName, String emailAddress, String contactNumber, String address, String suburb, String city, String postalCode, int activated) {

super(id, firstName, lastName, emailAddress, contactNumber, address, suburb, city, postalCode, activated);

}

public Member(int id, String birthDate, String address, String suburb, String city, String postalCode, String fistName, String lastName, String emailAddress, String contactNumber) {

super(id, address, suburb, city, postalCode, fistName, lastName, emailAddress, contactNumber);

this.birthdate = birthDate;

}

public Member() {

super();

}

public String getBirthdate() {

return birthdate;

}

public void setBirthdate(String birthdate) {

this.birthdate = birthdate;

}

@Override

public void Add() throws Exception {

MemberDAL.addMember(this);

}

@Override

public void Update() throws Exception {

MemberDAL.updateMember(this);

}

@Override

public void Deactivate(int id) throws Exception {

MemberDAL.deactivateMember(id);

}

@Override

public ArrayList<Person> Search() throws Exception {

return MemberDAL.getAllMember();

}

@Override

public ArrayList<Person> SearchByName(String name) throws Exception {

return MemberDAL.getMemberByName(name);

}

public ArrayList<String> SearchMemberName() throws Exception{

return MemberDAL.getAllMemberName();

}

}

**Person**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Model;

import java.util.ArrayList;

/\*\*

\*

\* @author Rom

\*/

public abstract class Person {

private int id;

private String firstName;

private String lastName;

private String emailAddress;

private String contactNumber;

private String address;

private String suburb;

private String city;

private String postalCode;

private int activated;

public Person(int id, String firstName, String lastName, String emailAddress, String contactNumber, String address, String suburb, String city, String postalCode, int activated) {

this.id = id;

this.firstName = firstName;

this.lastName = lastName;

this.emailAddress = emailAddress;

this.contactNumber = contactNumber;

this.address = address;

this.suburb = suburb;

this.city = city;

this.postalCode = postalCode;

this.activated = activated;

}

public Person(int id, String address, String suburb, String city, String postalCode, String firstName, String lastName, String emailAddress, String contactNumber){

this.id = id;

this.address = address;

this.suburb = suburb;

this.city = city;

this.postalCode = postalCode;

this.firstName = firstName;

this.lastName = lastName;

this.emailAddress = emailAddress;

this.contactNumber = contactNumber;

}

public Person(int id, String firstName, String lastName, String emailAddress, String contactNumber){

this.id = id;

this.firstName = firstName;

this.lastName = lastName;

this.emailAddress = emailAddress;

this.contactNumber = contactNumber;

}

public Person(){

}

public int getId() {

return id;

}

public String getFirstName() {

return firstName;

}

public void setFirstName(String firstName) {

this.firstName = firstName;

}

public String getLastName() {

return lastName;

}

public void setLastName(String lastName) {

this.lastName = lastName;

}

public String getEmailAddress() {

return emailAddress;

}

public void setEmailAddress(String emailAddress) {

this.emailAddress = emailAddress;

}

public String getContactNumber() {

return contactNumber;

}

public void setContactNumber(String contactNumber) {

this.contactNumber = contactNumber;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public String getSuburb() {

return suburb;

}

public void setSuburb(String suburb) {

this.suburb = suburb;

}

public String getCity() {

return city;

}

public void setCity(String city) {

this.city = city;

}

public String getPostalCode() {

return postalCode;

}

public void setPostalCode(String postalCode) {

this.postalCode = postalCode;

}

public int getActivated() {

return activated;

}

public void setActivated(int activated) {

this.activated = activated;

}

public abstract void Add() throws Exception;

public abstract void Update() throws Exception;

public abstract void Deactivate(int id) throws Exception;

public abstract ArrayList<Person> Search() throws Exception;

public abstract ArrayList<Person> SearchByName(String name) throws Exception;

}

**User**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Model;

import DAL.UserDAL;

import java.util.ArrayList;

/\*\*

\*

\* @author Rom

\*/

public class User extends Person {

private String username;

private String password;

public User(String username, String password, int id, String firstName, String lastName, String emailAddress, String contactNumber, String address, String suburb, String city, String postalCode, int activated) {

super(id, firstName, lastName, emailAddress, contactNumber, address, suburb, city, postalCode, activated);

this.username = username;

this.password = password;

}

public User(int id, String username, String password, String address, String suburb, String city, String postalCode, String firstName, String lastName, String emailAddress, String contactNumber) {

super(id, address, suburb, city, postalCode, firstName, lastName, emailAddress, contactNumber);

this.username = username;

this.password = password;

}

public User(String username, String password) {

this.username = username;

this.password = password;

}

public User(String username, int id, String firstName, String lastName, String emailAddress, String contactNumber) {

super(id, firstName, lastName, emailAddress, contactNumber);

this.username = username;

}

public User(String username){

this.username = username;

}

public User() {

}

public String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

@Override

public void Add() throws Exception {

UserDAL.addUser(this);

}

@Override

public void Update() throws Exception {

UserDAL.updateUser(this);

}

@Override

public void Deactivate(int id) throws Exception {

UserDAL.deactivateUser(id);

}

@Override

public ArrayList<Person> Search() throws Exception {

return UserDAL.getAllUser();

}

@Override

public ArrayList<Person> SearchByName(String name) throws Exception {

return UserDAL.getUserByName(name);

}

public void UpdatePassword(String username, String password, String newPassword) throws Exception{

UserDAL.updatePassword(username, password, newPassword);

}

public ArrayList<Person> CheckPassword(String username ,String password) throws Exception{

return UserDAL.checkPassword(username, password);

}

public void ResetPassword(int id)throws Exception{

UserDAL.resetPassword(id);

}

}

**Shared Package**

**SharedData**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Shared;

/\*\*

\*

\* @author massami

\*/

public class SharedData {

public static String username;

}

**Utility Package**

**GUIUtility**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Utility;

import java.util.Hashtable;

import javax.swing.JPanel;

import java.awt.Component;

import java.text.DateFormat;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.Date;

import java.util.Set;

import javax.swing.JComboBox;

import javax.swing.JTable;

import javax.swing.JTextField;

import javax.swing.table.DefaultTableModel;

import org.jdesktop.swingx.JXDatePicker;

/\*\*

\*

\* @author massami

\*/

public class GUIUtility {

/\*\*

\* Check if the mandatory field was filled

\*

\* @param data Evaluate the data

\* @return message containing errors

\*/

private static String checkInput(Hashtable data) {

String message = "";

Set<String> keys = data.keySet();

for (String k : keys) {

if (data.get(k).toString().length() == 0) {

message = message + "Please enter value for " + k + "\n";

}

}

return message;

}

/\*\*

\* Retrieve all controls that need to be validated

\*

\* @param panel control where the controls will be retrieved

\* @return data containing the controls

\*/

private static Hashtable<String, String> retrieveControls(JPanel panel) {

Hashtable<String, String> data = new Hashtable<String, String>();

for (Component c : panel.getComponents()) {

//Check TextField controls

if (c instanceof JTextField) {

JTextField j = (JTextField) c;

if (j.getName() != null && !j.getName().equals("")) {

data.put(j.getName(), j.getText());

}

}

//Check CheckBox controls

else if (c instanceof JComboBox) {

JComboBox cb = (JComboBox) c;

if (cb.getName() != null && !cb.getName().equals("")) {

data.put(cb.getName(), cb.getSelectedItem().toString());

}

}

//Check Date controls

else if (c instanceof JXDatePicker) {

JXDatePicker dp = (JXDatePicker) c;

if (dp.getName() != null && !dp.getName().equals("")) {

if (dp.getDate() != null) {

DateFormat format = new SimpleDateFormat("yyyy-MM-dd");

String date = format.format(dp.getDate());

data.put(dp.getName(), date);

} else {

data.put(dp.getName(), "");

}

}

}

}

return data;

}

/\*\*

\* Clean the components of the selected panel

\* @param panel

\*/

public static void cleanComponents(JPanel panel){

Hashtable<String, String> data = new Hashtable<String, String>();

for (Component c : panel.getComponents()) {

//Check TextField controls

if (c instanceof JTextField) {

JTextField j = (JTextField) c;

j.setText("");

}

//Check CheckBox controls

else if (c instanceof JComboBox) {

JComboBox cb = (JComboBox) c;

cb.setSelectedIndex(0);

}

//Check Date controls

else if (c instanceof JXDatePicker) {

JXDatePicker dp = (JXDatePicker) c;

dp.setDate(null);

}

}

}

/\*\*

\* Convert String to Date format

\* @param date String to be converted

\* @return formated String

\* @throws Exception

\*/

public static Date convertToDate(String date) throws Exception{

Calendar startDate = javax.xml.bind.DatatypeConverter.parseDateTime(date);

return startDate.getTime();

}

/\*\*

\* Convert Date to String

\* @param date Date to to be formated

\* @return formated String

\*/

public static String convertDateToString(Date date){

DateFormat format = new SimpleDateFormat("yyyy-MM-dd");

return format.format(date);

}

/\*\*

\* Check if the components are empty

\* @param panel return the components in the panel

\* @return message of validation

\*/

public static String validateInput(JPanel panel){

Hashtable<String, String> data = retrieveControls(panel);

String msg = checkInput(data);

return msg;

}

/\*\*

\* Clean the tables data

\*/

public static void cleanTable(JTable table) {

//Get the model of the table

DefaultTableModel model = (DefaultTableModel) table.getModel();

//Clean the table content

model.setRowCount(0);

}

}

**Library Package**

**LibrarySystem**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package librarysystem;

import GUI.Login;

/\*\*

\*

\* @author Rom

\*/

public class LibrarySystem {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

// TODO code application logic here

Login log = new Login();

log.setVisible(true);

}

}