

**NAME: PRIYANKA CHOWDHURY**  
**REG NO: 20BIT0197**

**TOPIC: LIBRARY MANAGEMENT SYSTEM**

**SUBJECT: JAVA PROGRAMMING**  
**COURSE CODE: CSE1007**

**FACULTY: DR. ANITHAA**

**Aim and Objective** – To create an Application for managing any Library with proper details of User(Name, Phone Number, Gender, etc), Book(Name, publisher, author), as well as issue and return functionality.

**Introduction-**

The library management system software helps in the reduction of manual labor. Manually managing a library is time-consuming and involves a massive quantity of paperwork. A computerized method eliminates the need of people and stationery. As a result, operating costs are reduced. The librarian can easily address questions about the availability of books. Adding, deleting, or changing records in the database is a straightforward procedure. It is simple to add new members or delete current ones. The library management system software makes the library smarter by organizing the books by author, title, and subject all having unique ids. This helps in quick and convenient search for books as per needs

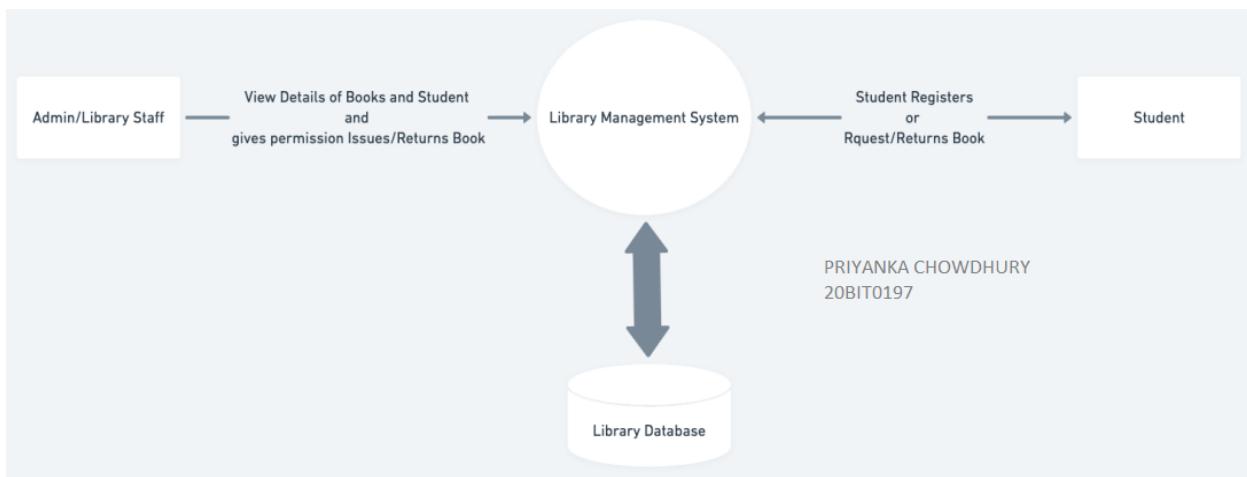
**H/W and S/W specifications-**

Hardware Specifications – HP laptop

Software Specifications –

- Compiler – IntelliJ Community
- Database – MySQL with WAMP

## Architecture - DFD -



## Modules –

1. Admin/Staff login: These are the ones who administer the system by adding or removing books into and from the system respectively and are also responsible for issuing and taking care of the return of books.
2. Add, Update and Delete Books: The admin/staff can add or delete books to the system by entering the details of the books and can even update the details.
3. Add, Update and Delete Users: The admin/staff can add or delete users to the system by entering the details of the user and can even update the details.
4. Issue: The admin/staff can view details for the book issuance.
5. Return: The admin/staff can view details for the book returns.
6. Display User
7. Display Available Books
8. Display borrowed Books along with the issue date.

## Code-

Login.java

```
package com.example.librarymanagementsystem;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
```

```
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.scene.control.Alert;
import javafx.scene.control.Button;
import javafx.scene.control.ButtonType;
import javafx.scene.control.TextField;
import javafx.scene.image.ImageView;
import javafx.scene.layout.Region;
import javafx.stage.Stage;

import java.io.IOException;

public class HelloController {
    @FXML
    private Button btn_login;

    @FXML
    private Button btn_register;

    @FXML
    private TextField tv_password;

    @FXML
    private TextField tv_username;

    @FXML
    private ImageView imageView;

    private FXMLLoader root;

    private Stage stage;

    private Scene scene;

    @FXML
    void onClickLogin(ActionEvent event) {
        String u = tv_username.getText();
        String p = tv_password.getText();
        if(u.isBlank() || p.isBlank()) {
```

```

        showMessage("User name or password can not be
blank",Alert.AlertType.NONE);
    }else if(p.length()<6){
        showMessage("Password can not be less than 6
characters",Alert.AlertType.NONE);
    }
    else {
        int val = DataBase.login(u,p);
        System.out.println(val);
        if (val != 0){
            root = new
FXMLLoader(getClass().getResource("home_page.fxml"));
            stage = (Stage) (btn_login.getScene().getWindow());
            try {
                stage.setScene(new Scene(root.load()));
            } catch (IOException e) {
                System.out.println(e);
            }
        }
        else{
            }
        }
    }
}

@FXML
void onClickRegister(ActionEvent event) {
    String u = tv_username.getText();
    String p = tv_password.getText();
    if(u.isBlank() || p.isBlank()){
        showMessage("User name or password can not be
blank",Alert.AlertType.NONE);
    }else if(p.length()<6){
        showMessage("Password can not be less than 6
characters",Alert.AlertType.NONE);
    }else {
        DataBase.register(u,p);
        root = new
FXMLLoader(getClass().getResource("home_page.fxml"));
        stage = (Stage) (btn_login.getScene().getWindow());
    }
}

```

```

        try {
            stage.setScene(new Scene(root.load()));
        } catch (IOException e) {
            System.out.println(e);
        }
    }

    private void showMessage(String content, Alert.AlertType type) {
        Alert alert = new Alert(type, content, ButtonType.OK);
        alert.getDialogPane().setMinHeight(Region.USE_PREF_SIZE);
        alert.show();
    }
}

```

## HomePage.java

```

package com.example.librarymanagementsystem;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.stage.Stage;

import java.io.IOException;

public class HomePage {

    @FXML
    private Button btn_addBooks;

    @FXML
    private Button btn_log;

    @FXML
    private Button btn_addCustomer;
}

```

```
@FXML
private Button btn_displayAvailable;

@FXML
private Button btn_displayBorrowed;

@FXML
private Button btn_displayCustomers;

@FXML
private Button btn_issueBook;

@FXML
private Button btn_returnBook;

private FXMLLoader root;

private Stage stage;

private Scene scene;

@FXML
void onClickAddBook(ActionEvent event) {
    root = new FXMLLoader(getClass().getResource("add_book.fxml"));
    stage = (Stage) (btn_addBooks.getScene().getWindow());
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}

@FXML
void onClickAddCustomer(ActionEvent event) {
    root = new
FXMLLoader(getClass().getResource("add_customer.fxml"));
    stage = (Stage) (btn_addCustomer.getScene().getWindow());
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
```

```
        System.out.println(e);
    }
}

@FXML
void onClickDisplayAvailable(ActionEvent event) {
    root = new
FXMLLoader(getClass().getResource("display_available.fxml"));
    stage = (Stage) (btn_displayAvailable.getScene().getWindow());
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}

@FXML
void onClickDisplayBorrowed(ActionEvent event) {
    root = new
FXMLLoader(getClass().getResource("display_borrowed.fxml"));
    stage = (Stage) (btn_displayBorrowed.getScene().getWindow());
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}

@FXML
void onClickLogOut(ActionEvent event) {
    root = new FXMLLoader(getClass().getResource("hello-view.fxml"));
    stage = (Stage) (btn_log.getScene().getWindow());
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}
```

```
@FXML
void onClickDisplayCustomer(ActionEvent event) {
    root = new
XMLLoader(getClass().getResource("display_customers.fxml"));
    stage = (Stage) (btn_displayCustomers.getScene().getWindow());
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}

@FXML
void onClickIssue(ActionEvent event) {
    root = new FXMLLoader(getClass().getResource("issue_book.fxml"));
    stage = (Stage) (btn_issueBook.getScene().getWindow());
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}

@FXML
void onClickReturn(ActionEvent event) {
    root = new FXMLLoader(getClass().getResource("return_book.fxml"));
    stage = (Stage) (btn_returnBook.getScene().getWindow());
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}

}
```

### AddBook.java

```
package com.example.librarymanagementsystem;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.scene.control.Alert;
import javafx.scene.control.Button;
import javafx.scene.control.ButtonType;
import javafx.scene.control.TextField;
import javafx.scene.layout.Region;
import javafx.stage.Stage;

import java.io.IOException;

public class AddBook {
    @FXML
    private Button btn_back;
    private FXMLLoader root;

    private Stage stage;

    private Scene scene;

    @FXML
    private Button btn_addBook;

    @FXML
    private TextField tv_authorName;

    @FXML
    private TextField tv_bookId;

    @FXML
    private TextField tv_bookName;

    @FXML
```

```

private TextField tv_publisherName;

@FXML
private Button btn_deleteBook;

@FXML
private Button btn_updateBook;

@FXML
void onClickDeleteBook(ActionEvent event) {
    int id = Integer.parseInt(tv_bookId.getText());
    DataBase.deleteBook(id);
}

@FXML
void onClickUpdateBook(ActionEvent event) {
    String name = tv_bookName.getText();
    String author = tv_authorName.getText();
    int id = Integer.parseInt(tv_bookId.getText());
    String pub = tv_publisherName.getText();
    DataBase.updateBook(id, name, author, pub);
}

@FXML
void onClickAddBook(ActionEvent event) {
    String name = tv_bookName.getText();
    String author = tv_authorName.getText();
    int id = Integer.parseInt(tv_bookId.getText());
    String pub = tv_publisherName.getText();
    if(name.isEmpty() || author.isEmpty()) {
        showMessage("All Fields are required", Alert.AlertType.ERROR);
    } else {
        DataBase.insertBook(id, name, author, pub);
    }
}

private static void showMessage(String content, Alert.AlertType type)
{
    Alert alert = new Alert(type, content, ButtonType.OK);
}

```

```

        alert.getDialogPane().setMinHeight(Region.USE_PREF_SIZE);
        alert.show();
    }

    @FXML
    void onClickBack(ActionEvent event) {
        root = new FXMLLoaderLoader(getClass().getResource("home_page.fxml"));
        stage = (Stage) (btn_back.getScene().getWindow());
        try {
            stage.setScene(new Scene(root.load()));
        } catch (IOException e) {
            System.out.println(e);
        }
    }
}

```

### AddUser.java

```

package com.example.librarymanagementsystem;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.scene.layout.Region;
import javafx.stage.Stage;

import java.io.IOException;

public class AddCustomer {
    @FXML
    private Button btn_back;
    private FXMLLoader root;

    private Stage stage;
    private Scene scene;

```

```
@FXML
void onClickBack(ActionEvent event) {
    root = new FXMLLoader(getClass().getResource("home_page.fxml"));
    stage = (Stage) btn_back.getScene().getWindow();
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}

@FXML
private Button btn_addCustomer;

@FXML
private ToggleGroup gender;

@FXML
private RadioButton tb_female;

@FXML
private RadioButton tb_male;

@FXML
private TextField tv_fName;

@FXML
private TextField tv_lName;

@FXML
private TextField tv_phoneNo;

@FXML
private TextField tv_userId;

@FXML
private Button btn_deleteUser;
```

```

private Button btn_updateUser;

private static void showMessage(String content, Alert.AlertType type)
{
    Alert alert = new Alert(type, content, ButtonType.OK);
    alert.getDialogPane().setMinHeight(Region.USE_PREF_SIZE);
    alert.show();
}

@FXML
void onClickAddCustomer(ActionEvent event) {
    String name = tv_fName.getText();
    String last = tv_lName.getText();
    int id = Integer.parseInt(tv_userId.getText());
    String gender = "";
    if(tb_male.isSelected()) {
        gender = "Male";
    } else {
        gender = "Female";
    }
    String phone = tv_phoneNo.getText();
    if(name.isEmpty() || last.isEmpty() ||
    tv_userId.getText().equals("")) {
        showMessage("All Feilds are required",
Alert.AlertType.ERROR);
    } else if(phone.length()!=10) {
        showMessage("Enter valid phone number", Alert.AlertType.NONE);
    }else{
        DataBase.insertUser(id,name,last,gender,phone);
    }
}

@FXML
void onClickDeleteCustomer(ActionEvent event) {
    int id = Integer.parseInt(tv_userId.getText());
    DataBase.deleteUser(id);
}

```

```

@FXML
void onClickUpdateCustomer(ActionEvent event) {
    int id = Integer.parseInt(tv_userId.getText());
    String name = tv_fName.getText();
    String last = tv_lName.getText();
    String gender = "";
    if(tb_male.isSelected()) {
        gender = "Male";
    }else {
        gender = "Female";
    }
    String phone = tv_phoneNo.getText();
    if(phone.length()!=10){
        showMessage("Enter valid phone number", Alert.AlertType.NONE);
    }else{
        DataBase.updateUser(id,name,last,gender,phone);
    }
}
}

```

### Borrowed.java

```

package com.example.librarymanagementsystem;

import java.sql.Date;

public class Borrowed {
    int uid,bid;
    String book;
    String user;
    Date date;

    Borrowed(){}
    Borrowed(int uid, String user,int bid,String book, Date date){
        this.uid = uid;
        this.bid = bid;
        this.book = book;
        this.user = user;
        this.date = date;
    }
}

```

```
}

public Date getDate() {
    return date;
}

public int getUid() {
    return uid;
}

public void setUid(int uid) {
    this.uid = uid;
}

public int getBid() {
    return bid;
}

public void setBid(int bid) {
    this.bid = bid;
}

public void setBook(String book) {
    this.book = book;
}

public String getBook() {
    return book;
}

public String getUser() {
    return user;
}

public void setUser(String user) {
    this.user = user;
}

}
```

### IssueBook.java

```
package com.example.librarymanagementsystem;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.DatePicker;
import javafx.scene.control.TextField;
import javafx.stage.Stage;

import java.io.IOException;
import java.sql.Date;

public class IssueBook {

    @FXML
    private Button btn_back;
    private FXMLLoader root;

    private Stage stage;

    private Scene scene;

    @FXML
    private Button btn_issueBook;

    @FXML
    private TextField tv_bookId;

    @FXML
    private TextField tv_userId;

    @FXML
    private DatePicker dp_date;

    @FXML
    void onClickIssueBook(ActionEvent event) {
        int bookId = Integer.parseInt(tv_bookId.getText());
    }
}
```

```

        int userId = Integer.parseInt(tv_userId.getText());
        Date date = Date.valueOf(dp_date.getValue());
        DataBase.issueBook(userId, bookId, date);
    }

    @FXML
    void onClickBack(ActionEvent event) {
        root = new FXMLLoader(getClass().getResource("home_page.fxml"));
        stage = (Stage) btn_back.getScene().getWindow();
        try {
            stage.setScene(new Scene(root.load()));
        } catch (IOException e) {
            System.out.println(e);
        }
    }
}

```

### ReturnBook.java

```

package com.example.librarymanagementsystem;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.TextField;
import javafx.stage.Stage;

import java.io.IOException;

public class ReturnBook {
    @FXML
    private Button btn_back;
    private FXMLLoader root;

    private Stage stage;
    private Scene scene;

    @FXML

```

```

private Button btn_returnBook;

@FXML
private TextField tv_bookId;

@FXML
private TextField tv_userId;

@FXML
void onClickReturnBook(ActionEvent event) {
    int bookId = Integer.parseInt(tv_bookId.getText());
    int userId = Integer.parseInt(tv_userId.getText());
    DataBase.returnBook(userId,bookId);
}

@FXML
void onClickBack(ActionEvent event) {
    root = new FXMLLoader(getClass().getResource("home_page.fxml"));
    stage = (Stage) (btn_back.getScene().getWindow());
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}
}

```

### DisplayUsers.java

```

package com.example.librarymanagementsystem;

import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.scene.Scene;

```

```
import javafx.scene.control.Button;
import javafx.scene.control.TableColumn;
import javafx.scene.control TableView;
import javafx.scene.control.TextField;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.stage.Stage;

import java.io.IOException;
import java.net.URL;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.ResourceBundle;

public class DisplayCustomers implements Initializable {

    ObservableList<User> items= FXCollections.observableArrayList();

    @FXML
    private Button btn_back;
    private FXMLLoader root;

    private Stage stage;

    private Scene scene;

    @FXML
    private Button btn_search;
    @FXML
    private TextField tv_search;
    @FXML
    void onClickSearch(ActionEvent event) {
        table_users.setItems(items);
        String search = tv_search.getText();
        tc_userId.setCellValueFactory(new PropertyValueFactory<User, Integer>("id"));
        tc_first.setCellValueFactory(new PropertyValueFactory<User, String>("name"));
    }
}
```

```

        tc_last.setCellValueFactory(new
PropertyValueFactory<User, String>("last"));
        tc_gender.setCellValueFactory(new
PropertyValueFactory<User, String>("gender"));
        tc_phone.setCellValueFactory(new
PropertyValueFactory<User, String>("phone"));
    try
    {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library","root","");
        PreparedStatement statement;
        if(search.isBlank()){
            statement = con.prepareStatement("select * from user");
        }else{
            statement = con.prepareStatement("select * from user where
fName = '"+ search +"'");
        }
        ResultSet rs = statement.executeQuery();
        items.clear();
        table_users.getItems().clear();

        while(rs.next()) {
            int s1 = rs.getInt(1);
            String s2 = rs.getString(2);
            String s3 = rs.getString(3);
            String s4 = rs.getString(4);
            String s5 = rs.getString(5);
            User stu =new User(s1, s2, s3, s4,s5);
            items.add(stu);
            System.out.println(s1 + " " + s2 + " " + s3 + " " + s4);
        }
        con.close();
    }catch(Exception e ){}
}

@FXML
private TableView<User> table_users;

```

```
@FXML
private TableColumn<User, String> tc_first;

@FXML
private TableColumn<User, String> tc_gender;

@FXML
private TableColumn<User, String> tc_last;

@FXML
private TableColumn<User, String> tc_phone;

@FXML
private TableColumn<User, Integer> tc_userId;

@FXML
void onClickBack(ActionEvent event) {
    root = new FXMLLoader(getClass().getResource("home_page.fxml"));
    stage = (Stage) btn_back.getScene().getWindow();
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}

@Override
public void initialize(URL location, ResourceBundle resources) {

    table_users.setItems(items);
    tc_userId.setCellValueFactory(new PropertyValueFactory<User, Integer>("id"));
    tc_first.setCellValueFactory(new PropertyValueFactory<User, String>("name"));
    tc_last.setCellValueFactory(new PropertyValueFactory<User, String>("last"));
    tc_gender.setCellValueFactory(new PropertyValueFactory<User, String>("gender"));
    tc_phone.setCellValueFactory(new PropertyValueFactory<User, String>("phone"));
}
```

```

try
{
    Class.forName("com.mysql.jdbc.Driver");
    Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library","root","");
);
    PreparedStatement statement = con.prepareStatement("select *
from user");
    ResultSet rs = statement.executeQuery();
    items.clear();
    table_users.getItems().clear();

    while(rs.next()) {
        int s1 = rs.getInt(1);
        String s2 = rs.getString(2);
        String s3 = rs.getString(3);
        String s4 = rs.getString(4);
        String s5 = rs.getString(5);
        User stu =new User(s1, s2, s3, s4,s5);
        items.add(stu);
        System.out.println(s1 + " " + s2 + " " + s3 + " " + s4);
    }
    con.close();
}catch(Exception e){}
}
}

```

### DisplayBorrowed.java

```

package com.example.librarymanagementsystem;

import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.scene.Scene;
import javafx.scene.control.Button;

```

```
import javafx.scene.control.TableColumn;
import javafx.scene.control.TableView;
import javafx.scene.control.TextField;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.stage.Stage;

import java.io.IOException;
import java.net.URL;
import java.sql.*;
import java.util.ResourceBundle;

public class DisplayBorrowed implements Initializable {
    ObservableList<Borrowed> items= FXCollections.observableArrayList();
    @FXML
    private Button btn_back;
    private FXMLLoader root;

    private Stage stage;

    private Scene scene;

    @FXML
    private Button btn_search;
    @FXML
    private TextField tv_search;
    @FXML
    void onClickSearch(ActionEvent event) {
        table_borrowed.setItems(items);
        String search = tv_search.getText();
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library","root","");
            PreparedStatement statement;
            if (search.isBlank()){
                statement = con.prepareStatement("select
user_id, fname, book_id, name from " +
                    "user natural join (borrowed natural join book)");
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```

        }else {
            statement = con.prepareStatement("select
user_id, fname, book_id, name, date from " +
                    "user natural join (borrowed natural join book)
where name = '" +search+"')";
        }
        ResultSet rs = statement.executeQuery();
        items.clear();
        table_borrowed.getItems().clear();
        tc_bookId.setCellValueFactory(new
PropertyValueFactory<Borrowed, Integer>("bid"));
        tc_bookName.setCellValueFactory(new
PropertyValueFactory<Borrowed, String>("book"));
        tc_studentId.setCellValueFactory(new
PropertyValueFactory<Borrowed, Integer>("uid"));
        tc_StudentName.setCellValueFactory(new
PropertyValueFactory<Borrowed, String>("user"));
        tc_date.setCellValueFactory(new
PropertyValueFactory<Borrowed, Date>("date"));

        while(rs.next()) {
            int s1 = rs.getInt(1);
            String s2 = rs.getString(2);
            int s3 = rs.getInt(3);
            String s4 = rs.getString(4);
            Date date = rs.getDate(5);
            Borrowed b =new Borrowed(s1, s2, s3, s4, date);
            items.add(b);
            System.out.println(s1 + " " + s2 + " " + s3 + " " + s4);
        }
        table_borrowed.setItems(items);
        con.close();
    }catch(Exception e){}
}

@FXML
private TableView<Borrowed> table_borrowed;

@FXML
private TableColumn<Borrowed, String> tc_StudentName;

```

```
@FXML
private TableColumn<Borrowed, Integer> tc_bookId;

@FXML
private TableColumn<Borrowed, Date> tc_date;

@FXML
private TableColumn<Borrowed, String> tc_bookName;

@FXML
private TableColumn<Borrowed, Integer> tc_studentId;

@FXML
void onClickBack(ActionEvent event) {
    root = new FXMLLoader(getClass().getResource("home_page.fxml"));
    stage = (Stage) btn_back.getScene().getWindow();
    try {
        stage.setScene(new Scene(root.load()));
    } catch (IOException e) {
        System.out.println(e);
    }
}

@Override
public void initialize(URL location, ResourceBundle resources) {
    table_borrowed.setItems(items);

    try {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library","root","");
        PreparedStatement statement = con.prepareStatement("select
user_id, fname, book_id, name, date from " +
                "user natural join (borrowed natural join book)");
        ResultSet rs = statement.executeQuery();
        items.clear();
    }
}
```

```

        table_borrowed.getItems().clear();
        tc_bookId.setCellValueFactory(new
PropertyValueFactory<Borrowed, Integer>("bid"));
        tc_bookName.setCellValueFactory(new
PropertyValueFactory<Borrowed, String>("book"));
        tc_studentId.setCellValueFactory(new
PropertyValueFactory<Borrowed, Integer>("uid"));
        tc_StudentName.setCellValueFactory(new
PropertyValueFactory<Borrowed, String>("user"));
        tc_date.setCellValueFactory(new
PropertyValueFactory<Borrowed, Date>("date"));

while(rs.next()) {
    int s1 = rs.getInt(1);
    String s2 = rs.getString(2);
    int s3 = rs.getInt(3);
    String s4 = rs.getString(4);
    Date date = rs.getDate(5);
    Borrowed b =new Borrowed(s1, s2, s3, s4, date);
    items.add(b);
    System.out.println(s1 + " " + s2 + " " + s3 + " " + s4 + "
" + date);
}
table_borrowed.setItems(items);
con.close();
}catch(Exception e ){
    System.out.println(e);
}
}
}
}

```

### Book.java

```

package com.example.librarymanagementsystem;

public class Book {
    private int id;
    private String name,author,pub;
    Book(){}
    Book(int id,String name, String author, String pub){

```

```
        this.id = id;
        this.name = name;
        this.author = author;
        this.pub = pub;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAuthor() {
        return author;
    }

    public void setAuthor(String last) {
        this.author = author;
    }

    public String getPub() {
        return pub;
    }

    public void setPub(String phone) {
        this.pub = pub;
    }
}
```

### User.java

```
package com.example.librarymanagementsystem;

public class User {
```

```
private int id;
private String name, last, gender, phone;

User() {}
User(int id, String name, String last, String gender, String phone) {
    this.id = id;
    this.name = name;
    this.last = last;
    this.gender = gender;
    this.phone = phone;
}

public int getId() {
    return id;
}
public void setId(int id) {
    this.id = id;
}

public String getName() {
    return name;
}
public void setName(String name) {
    this.name = name;
}

public String getLast() {
    return last;
}
public void setLast(String last) {
    this.last = last;
}

public String getPhone() {
    return phone;
}
public void setPhone(String phone) {
    this.phone = phone;
}
```

```

    public String getGender() {
        return gender;
    }
    public void setGender(String gender) {
        this.gender = gender;
    }
}

```

### Database.java

```

package com.example.librarymanagementsystem;
import javafx.scene.control.Alert;
import javafx.scene.control.ButtonType;
import javafx.scene.layout.Region;

import java.sql.*;
public class DataBase {

    private static void showMessage(String content, Alert.AlertType type)
    {
        Alert alert = new Alert(type, content, ButtonType.OK);
        alert.getDialogPane().setMinHeight(Region.USE_PREF_SIZE);
        alert.show();
    }
    public static void insertUser(int user_id, String fName, String lName,
String gender, String phone){
        boolean present = false;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",
"");
            Statement statement = con.createStatement();
            String S = "insert into user (user_id, fName, lName, gender,
mobile) values ('"+user_id+"','"+fName+"','"+lName+"','"+gender+"',
'"+phone+"')";
            System.out.println("Inserted Successfully!");
            statement.execute(S);
        }
    }
}

```

```

        } catch(SQLIntegrityConstraintViolationException sql) {
            present = true;
            showMessage("User already exists!", Alert.AlertType.ERROR);
        } catch (Exception e) {
            System.out.println(e);
        }
        if(present == false) {
            showMessage("Inserted Successfully!", Alert.AlertType.NONE);
        }
    }

    public static void updateUser(int user_id, String fName, String lName,
String gender , String phone){
    try {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",
"");
        Statement statement = con.createStatement();
        String S = "update user set fName = '" +fName+"',lName =
'"+lName+"', gender = '"+gender+"', mobile = '" +phone+ "' where user_id =
'"+user_id+"'";
        int i = statement.executeUpdate(S);

        if(i > 0) {
            System.out.println("Inserted Successfully!");
            showMessage("Updated Successfully!",
Alert.AlertType.NONE);
        } else {
            showMessage("User does not exist", Alert.AlertType.NONE);
        }
    } catch (Exception e) {
        System.out.println(e);
    }
}

    public static void updateBook(int id, String name, String auth, String
pub) {
    try {
        Class.forName("com.mysql.jdbc.Driver");

```

```

        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",
"");

        Statement statement = con.createStatement();
        String S = "update book set name = '" + name + "', author =
' " + auth + "' , publisher = '" + pub + "' where book_id = '" + id + "' ";
        int i = statement.executeUpdate(S);
        if(i > 0) {
            showMessage("Updated Successfully!",
Alert.AlertType.NONE);
            System.out.println("Inserted Successfully!");
        } else {
            showMessage("Book does not exist!", Alert.AlertType.NONE);
        }
    } catch (Exception e) {
        System.out.println(e);
    }
}

public static void deleteUser(int user_id){
try {
    Class.forName("com.mysql.jdbc.Driver");
    Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",
"");

    Statement statement = con.createStatement();
    String S = "delete from user where user_id = '" + user_id + "' ";
    int i = statement.executeUpdate(S);

    if(i > 0) {
        showMessage("Deleted Successfully!",
Alert.AlertType.NONE);
    } else {
        showMessage("User does not exist", Alert.AlertType.NONE);
    }
} catch (Exception e) {
    System.out.println(e);
}
}

```

```
public static void deleteBook(int book_id){  
    try {  
        Class.forName("com.mysql.jdbc.Driver");  
        Connection con =  
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",  
"");  
        Statement statement = con.createStatement();  
        String S = "delete from book where book_id = '" +book_id+"'" ;  
        int i = statement.executeUpdate(S);  
  
        if(i > 0) {  
            showMessage("Deleted Successfully!",  
Alert.AlertType.NONE);  
        } else {  
            showMessage("User does not exist", Alert.AlertType.NONE);  
        }  
    } catch (Exception e) {  
        System.out.println(e);  
    }  
}  
  
public static void insertBook(int book_id, String name, String author,  
String publisher){  
    boolean present = false;  
    try {  
        Class.forName("com.mysql.jdbc.Driver");  
        Connection con =  
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",  
"");  
        Statement statement = con.createStatement();  
        String inserts = "insert into book (book_id, name , author,  
publisher) values ('"+book_id+"','"+ name  
+"','"+author+"','"+publisher+"')";  
        statement.execute(inserts);  
    } catch(SQLIntegrityConstraintViolationException sql) {  
        showMessage("Book already inserted!", Alert.AlertType.ERROR);  
        present = true;  
    } catch(Exception e) {  
        System.out.println(e);  
    }  
}
```

```
        if(present == false) {
            System.out.println("Inserted Successfully!");
            showMessage("Inserted Successfully!", Alert.AlertType.NONE);
        }
    }

    public static void issueBook(int user_id, int book_id, Date date){
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",
"");
            Statement statement = con.createStatement();
            String s = "insert into borrowed (user_id,book_id, date)
values ('"+user_id+"','"+ book_id+"', '"+date+"')";
            System.out.println("Inserted Successfully!");
            showMessage("Book Issued!", Alert.AlertType.NONE);
            statement.execute(s);
        } catch (Exception e) {
            System.out.println(e);
        }
    }

    public static void returnBook(int user_id, int book_id){
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",
"");
            Statement statement = con.createStatement();
            String S = "delete from borrowed where book_id = '"+book_id+"'
and user_id = '"+user_id+"'";
            System.out.println("Deleted Successfully!");
            showMessage("Book Returned!", Alert.AlertType.NONE);
            statement.execute(S);
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

```
public static void register(String user_name, String password){  
    try{  
        Class.forName("com.mysql.jdbc.Driver");  
        Connection con =  
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",  
"");  
        Statement statement = con.createStatement();  
        String S = "insert into userLogin (user_name, password) values  
('"+user_name+"','"+password+"')";  
        System.out.println("Inserted Successfully!");  
        showMessage("Registered Successfully!", Alert.AlertType.NONE);  
        statement.execute(S);  
    }catch (Exception e){  
        System.out.println(e);  
    }  
}  
  
public static int login(String user_name , String password){  
    int present = 0;  
    try{  
        Class.forName("com.mysql.jdbc.Driver");  
        Connection con =  
DriverManager.getConnection("jdbc:mysql://localhost:3306/library", "root",  
"");  
//        Statement statement = con.createStatement();  
        PreparedStatement statement = con.prepareStatement("select *  
from userLogin where user_name = '"+user_name+"' and password = '"+  
password+"'");  
        ResultSet rs = statement.executeQuery();  
        int count = 0;  
        while(rs.next()) {  
            count++;  
        }  
//        System.out.println(rs.next());  
        if(count==0){  
            showMessage("Invalid credentials", Alert.AlertType.NONE);  
            present =0;  
        }else{  
            showMessage("Logged In", Alert.AlertType.NONE);  
        }  
    }  
}
```

```

        present = 1;
    }
} catch (Exception e) {
    System.out.println(e);
}
return present;
}

public static void displayUser() {
}
}

```

## FXML CODES -

```

<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>

<!-->

<?import javafx.scene.control.PasswordField?>
<AnchorPane prefHeight="400" prefWidth="800" style="
-fx-border-color:#6A5ACD; -fx-border-width: 5; -fx-border-style: solid;" styleClass="root" stylesheets="@style.css"
xmlns="http://javafx.com/javafx/17" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.librarymanagementsystem.HelloController">
    <children>
        <Label layoutX="168.0" layoutY="38.0" prefHeight="56.0"
prefWidth="634.0" text="Library Management System" textAlignment="CENTER"
textFill="DARKSLATEBLUE">
            <font>
                <Font name="Bell MT Bold" size="48.0" />

```

```

        </font>
    </Label>
    <Label layoutX="40.0" layoutY="177.0" prefHeight="28.0"
prefWidth="164.0" text="UserName:>
    <font>
        <Font name="Calibri Bold" size="24.0" />
    </font></Label>
    <Label layoutX="43.0" layoutY="212.0" text="Password: ">
    <font>
        <Font name="Calibri Bold" size="24.0" />
    </font></Label>
    <TextField fx:id="tv_username" layoutX="172.0" layoutY="178.0"
prefHeight="26.0" prefWidth="221.0" />
    <PasswordField fx:id="tv_password" layoutX="172.0" layoutY="213.0"
prefHeight="26.0" prefWidth="222.0" />
    <Button fx:id="btn_register" layoutX="43.0" layoutY="292.0"
mnemonicParsing="false" onAction="#onClickRegister" prefHeight="38.0"
prefWidth="164.0" style="-fx-border-color: #6A5ACD; -fx-border-radius: 5;
-fx-background-color: #6A5ACD;" text="REGISTER" textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font></Button>
    <Button fx:id="btn_login" layoutX="230.0" layoutY="292.0"
mnemonicParsing="false" onAction="#onClickLogin" prefHeight="38.0"
prefWidth="164.0" style="-fx-border-color: #6A5ACD; -fx-border-radius: 5;
-fx-background-color: #6A5ACD;" text="LOGIN" textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font></Button>
    <ImageView fx:id="imageView" fitHeight="202.0" fitWidth="164.0"
layoutX="695.0" layoutY="241.0" pickOnBounds="true" preserveRatio="true">
    </ImageView>
    <ImageView fitHeight="293.0" fitWidth="423.0" layoutX="421.0"
layoutY="141.0" pickOnBounds="true" preserveRatio="true">
        <image>
            <Image url="@library-20clipart-clip-art-library-500_345.png"
/>
        </image></ImageView>
    </children>
</AnchorPane>

```

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.Cursor?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>

<AnchorPane prefHeight="516.0" prefWidth="924.0" style="
-fx-border-color:#6A5ACD; -fx-border-width: 5; -fx-border-style: solid;" styleClass="root" stylesheets="@style.css"
xmlns="http://javafx.com/javafx/17" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.librarymanagementsystem.HomePage">
    <children>
        <Label alignment="CENTER" layoutX="99.0" layoutY="14.0"
prefHeight="73.0" prefWidth="690.0" text="Welcome to XYZ Library"
textFill="DARKSLATEBLUE">
            <font>
                <Font name="Californian FB Bold" size="53.0" />
            </font>
        </Label>
        <Button fx:id="btn_addCustomer" layoutX="336.0" layoutY="121.0"
mnemonicParsing="false" onAction="#onClickAddCustomer"
style="-fx-border-radius: 5; -fx-border-color: #6A5ACD;
-fx-background-color: #FFF; -fx-background-color: #6A5ACD;" text="ADD/UPDATE/DELETE USER" textFill="WHITE">
            <font>
                <Font name="System Bold" size="16.0" />
            </font></Button>
        <Button fx:id="btn_issueBook" layoutX="319.0" layoutY="238.0"
mnemonicParsing="false" onAction="#onClickIssue" style="-fx-border-radius:
5; -fx-border-color: #6A5ACD; -fx-background-color: #FFF;
-fx-background-color: #6A5ACD;" text="ISSUE BOOK" textFill="WHITE">
            <font>
                <Font name="System Bold" size="16.0" />
            </font></Button>
```

```
<Button fx:id="btn_addBooks" layoutX="333.0" layoutY="175.0"
mnemonicParsing="false" onAction="#onClickAddBook"
style="-fx-border-radius: 5; -fx-border-color: #6A5ACD;
-fx-background-color: #FFF; -fx-background-color: #6A5ACD;" text="ADD/UPDATE/DELETE BOOK" textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font></Button>
<Button fx:id="btn_returnBook" layoutX="451.0" layoutY="238.0"
mnemonicParsing="false" onAction="#onClickReturn"
style="-fx-border-radius: 5; -fx-border-color: #6A5ACD;
-fx-background-color: #FFF; -fx-background-color: #6A5ACD;" text="RETURN
BOOK" textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font></Button>
<Button fx:id="btn_displayCustomers" layoutX="394.0" layoutY="316.0"
mnemonicParsing="false" onAction="#onClickDisplayCustomer"
style="-fx-border-radius: 50; -fx-background-radius: 50; -fx-border-color:
DARKSLATEBLUE; -fx-background-color: DARKSLATEBLUE;" text="DISPLAY USERS"
textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font>
    <cursor>
        <Cursor fx:constant="DEFAULT" />
    </cursor>
</Button>
<Button fx:id="btn_displayAvailable" layoutX="114.0" layoutY="316.0"
mnemonicParsing="false" onAction="#onClickDisplayAvailable"
style="-fx-border-radius: 50; -fx-background-radius: 50; -fx-border-color:
DARKSLATEBLUE; -fx-background-color: DARKSLATEBLUE;" text="DISPLAY
AVAILABLE BOOKS" textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font>
    <cursor>
        <Cursor fx:constant="DEFAULT" />
    </cursor>
</Button>
```

```

<Button fx:id="btn_displayBorrowed" layoutX="588.0" layoutY="316.0"
mnemonicParsing="false" onAction="#onClickDisplayBorrowed"
style="-fx-border-radius: 50; -fx-background-radius: 50; -fx-border-color:
DARKSLATEBLUE; -fx-background-color: DARKSLATEBLUE;" text="DISPLAY
BORROWED BOOKS" textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font>
    <cursor>
        <Cursor fx:constant="DEFAULT" />
    </cursor>
</Button>
<Button fx:id="btn_log" layoutX="773.0" layoutY="421.0"
mnemonicParsing="false" onAction="#onClickLogOut"
style="-fx-border-radius: 5; -fx-border-color: #6A5ACD;
-fx-background-color: #FFF; -fx-background-color: #6A5ACD;" text="LOG OUT"
textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font>
</Button>
</children>
</AnchorPane>

```

```

<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>

<AnchorPane prefHeight="400" prefWidth="800" style="
-fx-border-color:#6A5ACD; -fx-border-width: 5; -fx-border-style: solid;" styleClass="root" stylesheets="@style.css"

```

```
xmlns="http://javafx.com/javafx/17" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.librarymanagementsystem.AddBook">
    <children>
        <Button fx:id="btn_back" layoutX="822.0" layoutY="443.0"
mnemonicParsing="false" onAction="#onClickBack" style="-fx-border-color:
#6A5ACD; -fx-background-color: #6A5ACD; -fx-border-width: 1;
-fx-border-radius: 5;" text="BACK" textFill="WHITE">
            <font>
                <Font name="System Bold" size="16.0" />
            </font></Button>
        <Label layoutX="361.0" layoutY="39.0" text="ADD BOOK"
textFill="DARKSLATEBLUE">
            <font>
                <Font name="Bell MT Bold" size="36.0" />
            </font>
        </Label>
        <Label layoutX="132.0" layoutY="143.0" text="Book Name:>
            <font>
                <Font name="Calibri Bold" size="24.0" />
            </font>
        </Label>
        <Label layoutX="132.0" layoutY="180.0" text="Author Name:>
            <font>
                <Font name="Calibri Bold" size="24.0" />
            </font>
        </Label>
        <Label layoutX="132.0" layoutY="216.0" text="Publisher Name:>
            <font>
                <Font name="Calibri Bold" size="24.0" />
            </font>
        </Label>
        <TextField fx:id="tv_bookName" layoutX="351.0" layoutY="145.0" />
        <TextField fx:id="tv_authorName" layoutX="351.0" layoutY="181.0" />
        <TextField fx:id="tv_publisherName" layoutX="351.0" layoutY="217.0"
/>
        <Button fx:id="btn_addBook" layoutX="113.0" layoutY="328.0"
mnemonicParsing="false" onAction="#onClickAddBook"
style="-fx-border-color: #6A5ACD; -fx-background-color: #6A5ACD;
-fx-border-width: 1; -fx-border-radius: 5;" text="ADD BOOK"
textFill="WHITE">
```

```

<font>
    <Font name="System Bold" size="16.0" />
</font></Button>
<Label layoutX="132.0" layoutY="107.0" text="Book Id: ">
    <font>
        <Font name="Calibri Bold" size="24.0" />
    </font>
</Label>
<TextField fx:id="tv_bookId" layoutX="351.0" layoutY="109.0" />
<Button fx:id="btn_updateBook" layoutX="251.0" layoutY="328.0"
mnemonicParsing="false" onAction="#onClickUpdateBook"
style="-fx-border-color: #6A5ACD; -fx-background-color: #6A5ACD;
-fx-border-width: 1; -fx-border-radius: 5;" text="UPDATE BOOK"
textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font></Button>
<Button fx:id="btn_deleteBook" layoutX="427.0" layoutY="328.0"
mnemonicParsing="false" onAction="#onClickDeleteBook"
style="-fx-border-color: #6A5ACD; -fx-background-color: #6A5ACD;
-fx-border-width: 1; -fx-border-radius: 5;" text="DELETE BOOK"
textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font></Button>
<ImageView fitHeight="262.0" fitWidth="267.0" layoutX="600.0"
layoutY="105.0" pickOnBounds="true" preserveRatio="true">
    <image>
        <Image
url="@istockphoto-950962136-612x612-removebg-preview.png" />
    </image>
</ImageView>
</children>
</AnchorPane>

<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>

```

```

<?import javafx.scene.control.TableView>
<?import javafx.scene.control.TextField>
<?import javafx.scene.layout.AnchorPane>
<?import javafx.scene.text.Font>

<AnchorPane prefHeight="400" prefWidth="800" style="
-fx-border-color:#6A5ACD; -fx-border-width: 5; -fx-border-style: solid;" styleClass="root" stylesheets="@style.css"
xmlns="http://javafx.com/javafx/17" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.librarymanagementsystem.DisplayBorrowed">
    <children>
        <Button fx:id="btn_back" layoutX="821.0" layoutY="447.0" mnemonicParsing="false" onAction="#onClickBack" style="-fx-border-color:#6A5ACD; -fx-border-radius: 5; -fx-background-color: #6A5ACD;" text="BACK" textFill="WHITE">
            <font>
                <Font name="System Bold" size="16.0" />
            </font></Button>
        <Label layoutX="279.0" layoutY="49.0" text="BORROWED BOOKS" textFill="DARKSLATEBLUE">
            <font>
                <Font name="Californian FB Bold" size="36.0" />
            </font>
        </Label>
        <TableView fx:id="table_borrowed" layoutX="77.0" layoutY="116.0" prefHeight="200.0" prefWidth="746.0">
            <columns>
                <TableColumn fx:id="tc_studentId" prefWidth="127.19998168945312" text="USER ID" />
                <TableColumn fx:id="tc_StudentName" prefWidth="182.4000244140625" text="USER NAME" />
                <TableColumn fx:id="tc_bookId" prefWidth="120.80007934570312" text="BOOK ID" />
                <TableColumn fx:id="tc_bookName" prefWidth="211.99993896484375" text="BOOK NAME" />
                <TableColumn fx:id="tc_date" prefWidth="100.0" text="ISSUE DATE" />
            </columns>
        </TableView>
    </children>

```

```

        <TextField fx:id="tv_search" layoutX="132.0" layoutY="363.0"
prefHeight="38.0" prefWidth="298.0" promptText="Enter name of borrowed
book..." />
        <Button fx:id="btn_search" layoutX="458.0" layoutY="363.0"
mnemonicParsing="false" onAction="#onClickSearch" style="-fx-border-color:
#6A5ACD; -fx-border-radius: 5; -fx-background-color: #6A5ACD;" text="SEARCH" textFill="WHITE">
        <font>
            <Font name="System Bold" size="16.0" />
        </font></Button>
    </children>
</AnchorPane>
```

```

<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>

<AnchorPane prefHeight="400" prefWidth="800" style="
-fx-border-color:#6A5ACD; -fx-border-width: 5; -fx-border-style: solid;" styleClass="root" stylesheets="@style.css"
xmlns="http://javafx.com/javafx/17" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.librarymanagementsystem.DisplayCustomers">
    <children>
        <Button fx:id="btn_back" layoutX="822.0" layoutY="448.0"
mnemonicParsing="false" onAction="#onClickBack" style="-fx-border-color:
#6A5ACD; -fx-border-radius: 5; -fx-background-color: #6A5ACD;" text="BACK"
textFill="WHITE">
        <font>
            <Font name="System Bold" size="16.0" />
        </font></Button>
        <Label layoutX="404.0" layoutY="47.0" text="USERS"
textFill="DARKSLATEBLUE">
        <font>
```

```

        <Font name="Californian FB Bold" size="36.0" />
    </font>
</Label>
<TableView fx:id="table_users" layoutX="59.0" layoutY="108.0"
prefHeight="231.0" prefWidth="802.0">
    <columns>
        <TableColumn fx:id="tc_userId" prefWidth="97.5999984741211"
text="USER ID" />
        <TableColumn fx:id="tc_first" prefWidth="214.39996337890625"
text="FIRST NAME" />
        <TableColumn fx:id="tc_last" prefWidth="180.80007934570312"
text="LAST NAME" />
        <TableColumn fx:id="tc_gender" prefWidth="121.5999755859375"
text="GENDER" />
        <TableColumn fx:id="tc_phone" prefWidth="191.2000732421875"
text="PHONE NO" />
    </columns>
</TableView>
<TextField fx:id="tv_search" layoutX="59.0" layoutY="364.0"
prefHeight="38.0" prefWidth="370.0" promptText="Enter name to search user"
/>
<Button fx:id="btn_search" layoutX="460.0" layoutY="364.0"
mnemonicParsing="false" onAction="#onClickSearch" style="-fx-border-color:
#6A5ACD; -fx-border-radius: 5; -fx-background-color: #6A5ACD;"
text="SEARCH" textFill="WHITE">
    <font>
        <Font name="System Bold" size="16.0" />
    </font></Button>
</children>
</AnchorPane>

```

```

<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.control.TextField?>
```

```

<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>

<AnchorPane prefHeight="400" prefWidth="800" style="
-fx-border-color:#6A5ACD; -fx-border-width: 5; -fx-border-style: solid;" styleClass="root" stylesheets="@style.css"
xmlns="http://javafx.com/javafx/17" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.librarymanagementsystem.DisplayAvailable">
    <children>
        <Button fx:id="btn_back" layoutX="827.0" layoutY="449.0"
mnemonicParsing="false" onAction="#onClickBack" style="-fx-border-color:
#48 #6A5ACD3D8B; -fx-border-radius: 5; -fx-background-color: #6A5ACD;" text="BACK" textFill="WHITE">
            <font>
                <Font name="System Bold" size="16.0" />
            </font></Button>
        <Label layoutX="288.0" layoutY="49.0" text="AVAILABLE BOOKS"
textFill="DARKSLATEBLUE">
            <font>
                <Font name="Californian FB Bold" size="36.0" />
            </font>
        </Label>
        <TableView fx:id="tv_tableBooks" layoutX="133.0" layoutY="118.0"
prefHeight="200.0" prefWidth="686.0">
            <columns>
                <TableColumn fx:id="tc_bookId" prefWidth="136.80001831054688"
text="BOOK ID" />
                <TableColumn fx:id="tc_bookName" prefWidth="195.20001220703125"
text="BOOK NAME" />
                <TableColumn fx:id="tc_author" minWidth="0.0"
prefWidth="172.80001831054688" text="AUTHOR" />
                <TableColumn fx:id="tc_publisher" minWidth="0.0"
prefWidth="182.39996337890625" text="PUBLISHER" />
            </columns>
        </TableView>
        <TextField fx:id="tv_search" layoutX="137.0" layoutY="365.0"
prefHeight="38.0" prefWidth="312.0" promptText="Enter name of book" />
        <Button fx:id="btn_search" layoutX="465.0" layoutY="365.0"
mnemonicParsing="false" onAction="#onClickSearch" style="-fx-border-color:

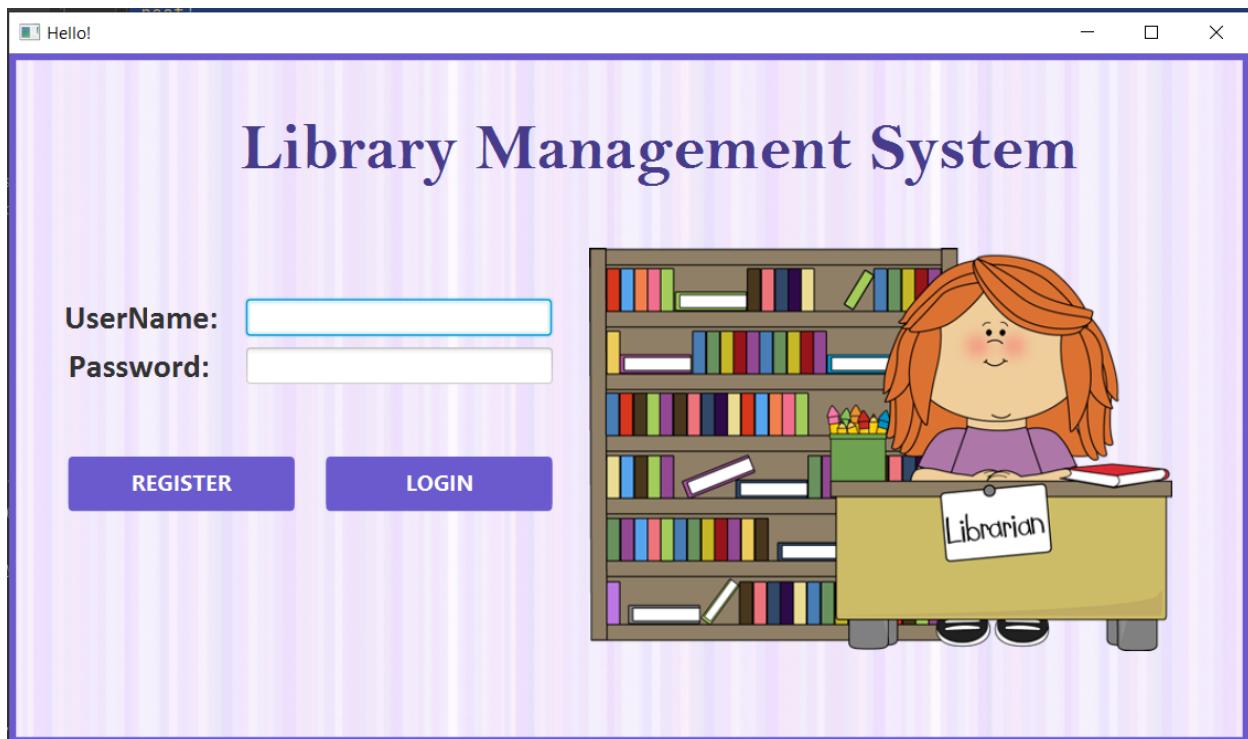
```

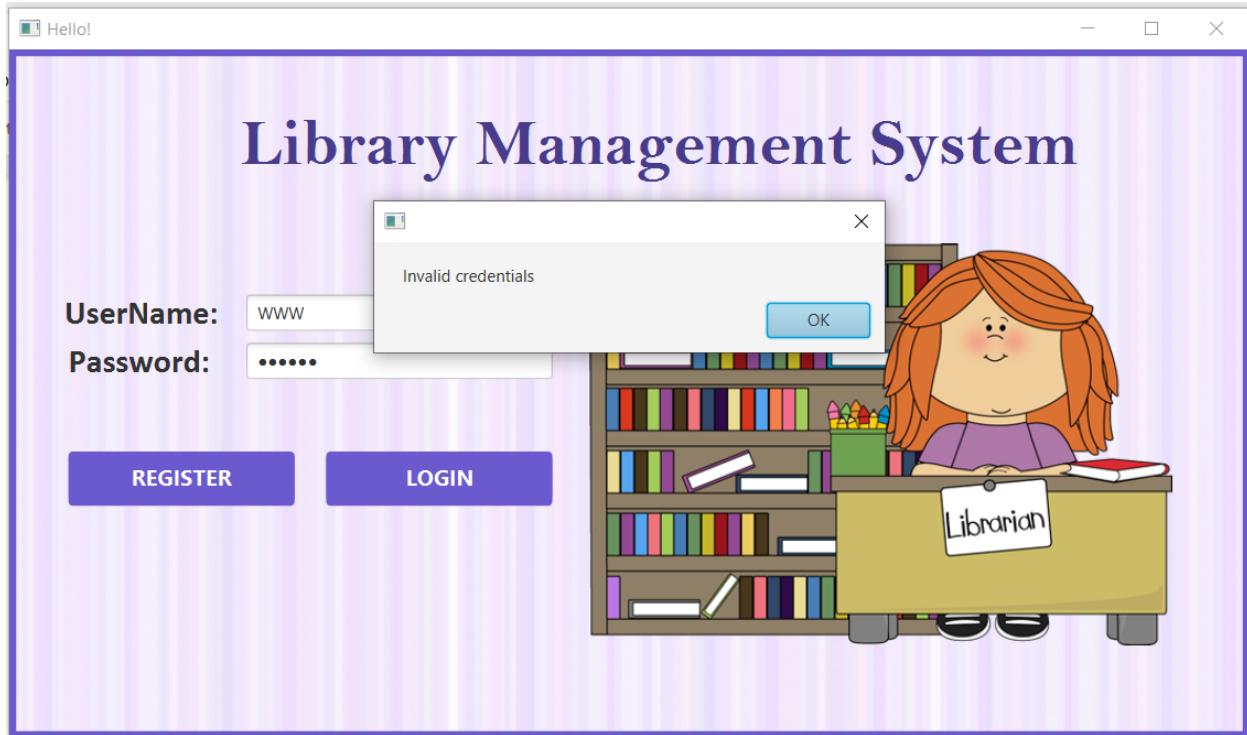
```
#48 #6A5ACD3D8B; -fx-border-radius: 5; -fx-background-color: #6A5ACD;"  
text="SEARCH" textFill="WHITE">  
    <font>  
        <Font name="System Bold" size="16.0" />  
    </font></Button>  
</children>  
</AnchorPane>
```

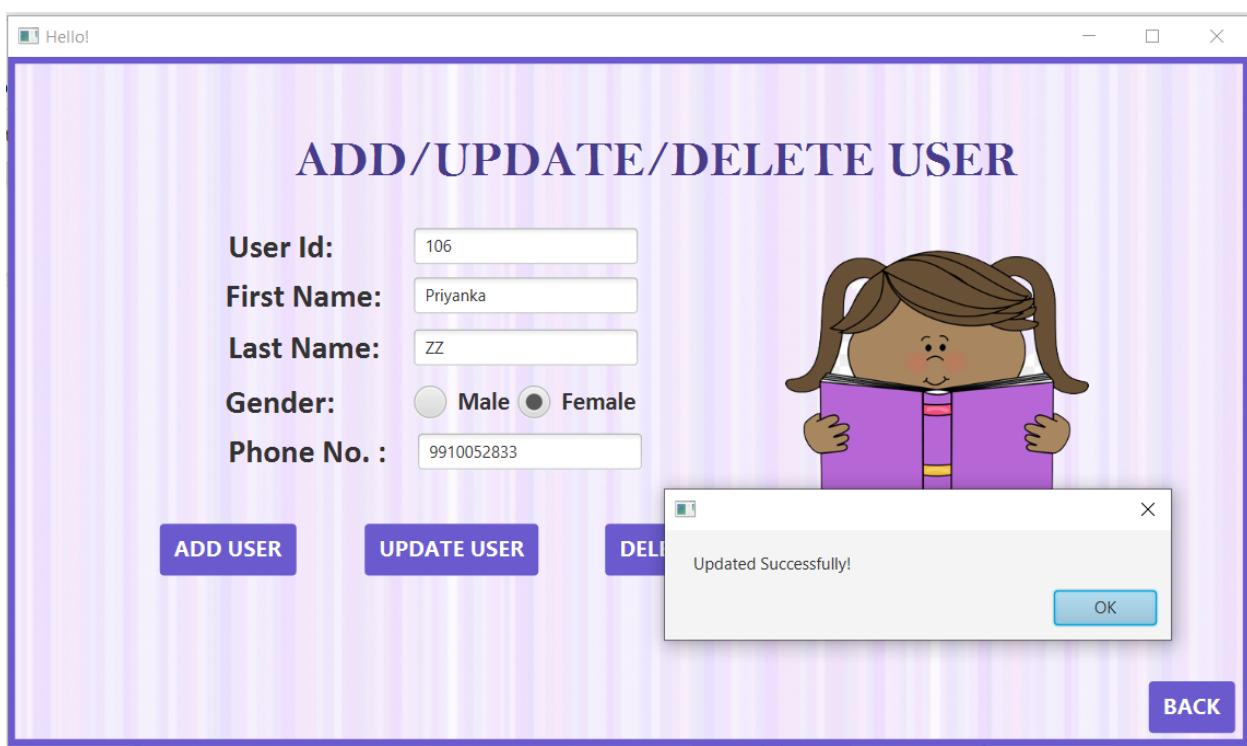
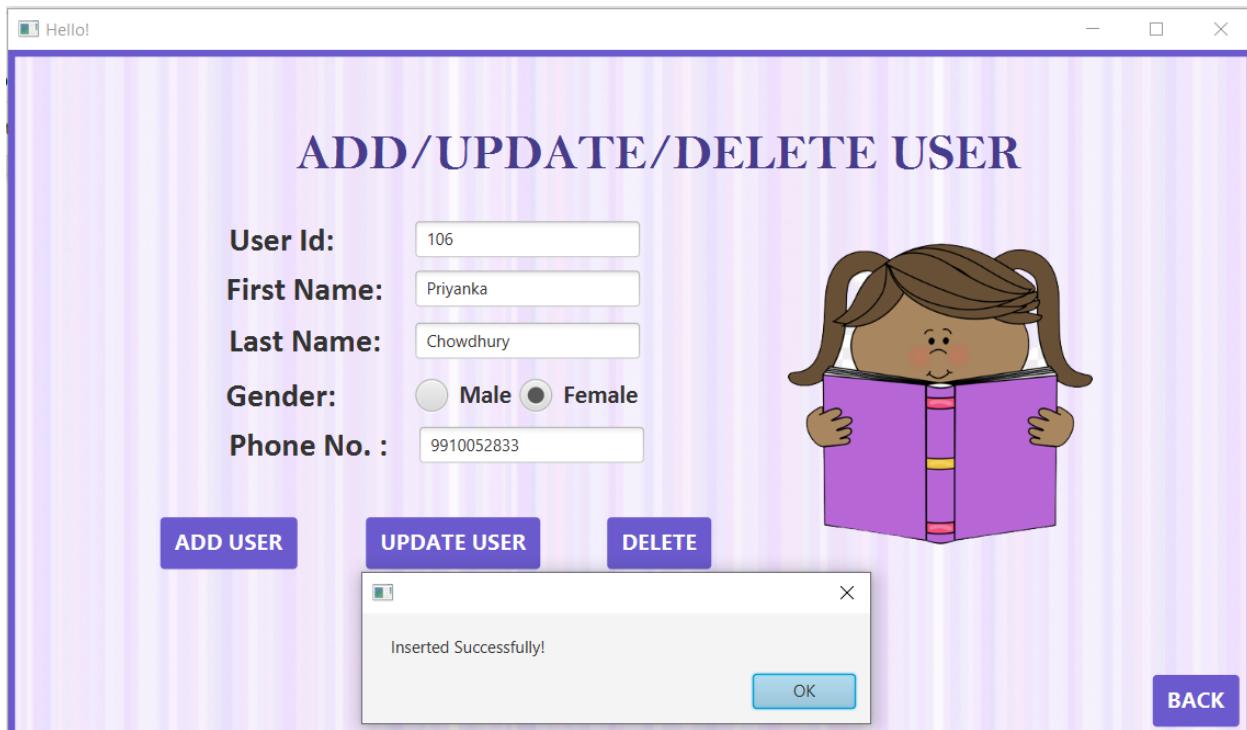
### Style.css

```
.root{  
    -fx-background-image: url('new.jpg');  
    -fx-background-size: cover;  
}
```

### OUTPUT SS -







Hello!

## ADD BOOK

**Book Id:**

**Book Name:**

**Author Name:**

**Publisher Name:**

**ADD BOOK**   **UPDATE BOOK**   **DELETE BOOK**



**BACK**

Hello!

## ISSUE BOOK

**Enter Book ID:**

**Enter User ID:**

**Date:**  2022

<	April	>	<	2022	>	
Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7



**BACK**

Hello!

## RETURN BOOK

Enter Book ID:

Enter User ID:

**RETURN BOOK**

Please Return Library Books Here

**BACK**

Hello!

## AVAILABLE BOOKS

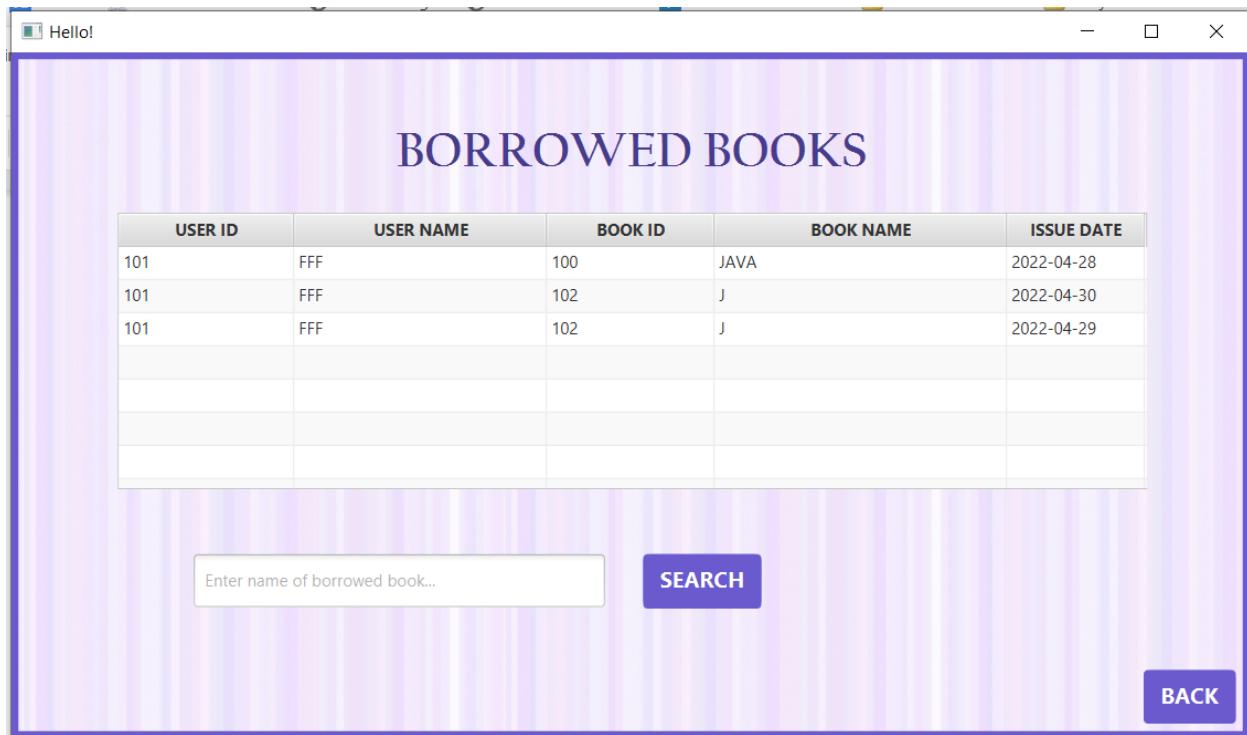
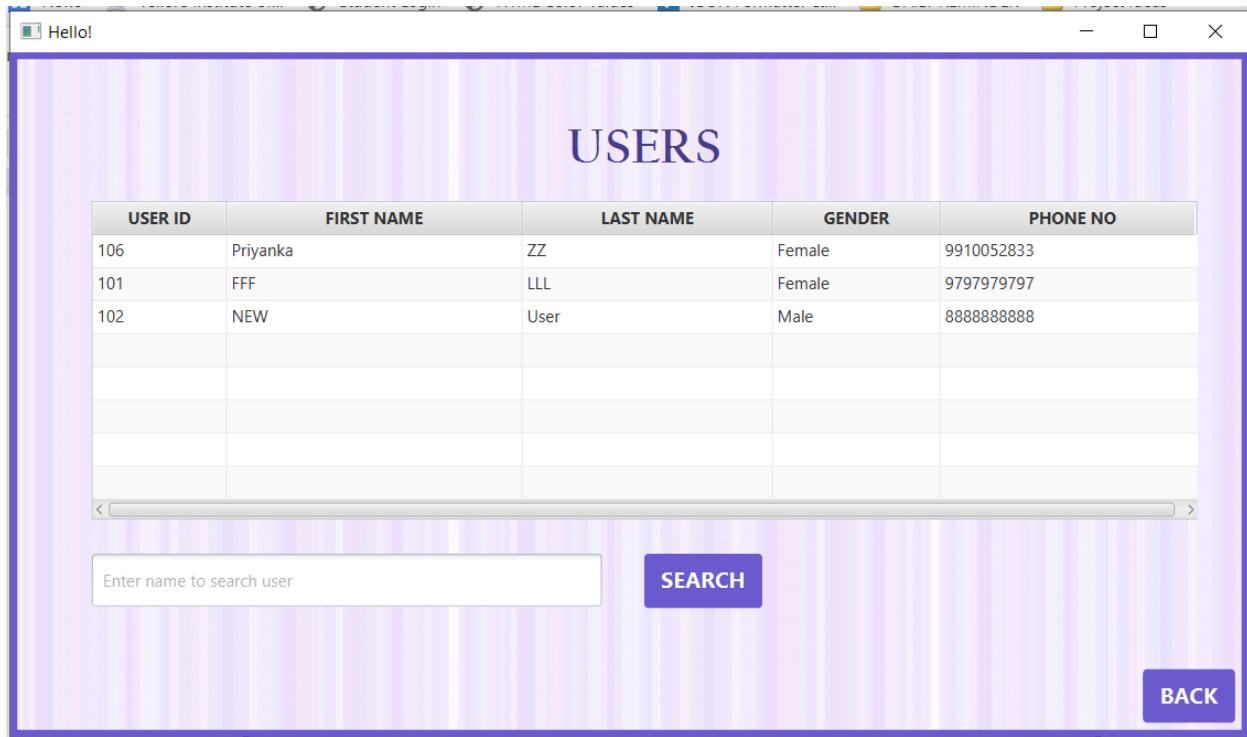
BOOK ID	BOOK NAME	AUTHOR	PUBLISHER
100	JAVA	XXX	XYZ
101	TOC	AAA	XXX
102	J	J	J

< >

Enter name of book

**SEARCH**

**BACK**



**Result and discussion -**

We get a fully functional Library Management Application built on JavaFX using MySQL Database and WAMP. This will make it easier to manage any institution's library.

**Conclusion -** Using this application, institutions will be able to maintain their libraries in a more efficient manner and reduce manual labor.