**Chapter 4:**

**SYSTEM IMPLEMENTATION**

**4.1 Tools and languages used:**

**4.1.1 Tools used:**

1. IntelliJ IDEA IDE Community Edition 2018.2 IDE
2. Java Development Kit 10.0.2
3. Scenebuilder 2.0
4. MySQL 8.0 Client and Shell

**4.1.2 Languages used:**

1. Java
2. JavaFX Framework
3. SQL (Structured Query Language)

**4.2 Source Code**

**4.2.0 Establishing Connection to MySQL Database:**

package sample;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class DBConnection {

public static Connection getConnection() throws SQLException {

String dbName="restaurantdb";

String username="root";

String password="toor";

Connection connection= DriverManager.getConnection("jdbc:mysql://localhost/"+dbName,username,password);

return connection;

}

}

**4.2.1 Login Page:**

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

<?import com.jfoenix.controls.JFXButton?>

<?import com.jfoenix.controls.JFXPasswordField?>

<?import com.jfoenix.controls.JFXTextField?>

<?import de.jensd.fx.glyphs.fontawesome.FontAwesomeIcon?>

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

<?import javafx.scene.image.Image?>

<?import javafx.scene.image.ImageView?>

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

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

<AnchorPane prefHeight="500.0" prefWidth="800.0" style="-fx-background-color: #FFFF;" xmlns="http://javafx.com/javafx/10.0.1" xmlns:fx="http://javafx.com/fxml/1" fx:controller="sample.Controller">

<children>

<ImageView fitHeight="500.0" fitWidth="286.0" pickOnBounds="true">

<image>

<Image url="@cover.jpg" />

</image>

</ImageView>

<Label layoutX="343.0" layoutY="172.0" prefHeight="64.0" prefWidth="267.0" text="Login">

<font>

<Font name="System Bold Italic" size="65.0" />

</font>

</Label>

<JFXTextField fx:id="uname" layoutX="343.0" layoutY="276.0" prefHeight="35.0" prefWidth="279.0" promptText="Enter Username" />

<JFXPasswordField fx:id="pass" layoutX="343.0" layoutY="363.0" prefHeight="26.0" prefWidth="279.0" promptText="Enter Password" />

<JFXButton fx:id="login\_btn" contentDisplay="GRAPHIC\_ONLY" layoutX="343.0" layoutY="428.0" onAction="#login" prefHeight="62.0" prefWidth="101.0">

<graphic>

<FontAwesomeIcon glyphName="ARROW\_CIRCLE\_RIGHT" size="40" />

</graphic>

</JFXButton>

<ImageView fitHeight="172.0" fitWidth="514.0" layoutX="286.0" pickOnBounds="true">

<image>

<Image url="@logo.jpg" />

</image>

</ImageView>

</children>

</AnchorPane>

**4.2.2 Backend Validation:**

import java.io.IOException;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

public class Controller {

@FXML

private JFXTextField uname;

@FXML

private JFXPasswordField pass;

@FXML

private JFXButton login\_btn;

@FXML

public void login(ActionEvent actionEvent) throws SQLException, ClassNotFoundException, IOException {

Connection con=DBConnection.getConnection();

String sql="select \* from users where username='"+uname.getText()+"' and password='"+pass.getText()+"'";

PreparedStatement ps=con.prepareStatement(sql);

ResultSet rs= ps.executeQuery();

if (rs.next())

{

Stage stage = (Stage) login\_btn.getScene().getWindow();

AnchorPane root;

root = (AnchorPane) FXMLLoader.load(getClass().getResource("dashboard.fxml"));

Scene scene = new Scene(root);

stage.setScene(scene);

}

else

{

Alert errorAlert = new Alert(Alert.AlertType.ERROR);

errorAlert.setHeaderText("Authorisation Error!");

errorAlert.setContentText("Invalid Username/Password");

errorAlert.showAndWait();

}

}

}

**4.2.3 Adding Dishes:**

package sample;

import com.jfoenix.controls.JFXButton;

import com.jfoenix.controls.JFXComboBox;

import com.jfoenix.controls.JFXTextField;

import javafx.collections.FXCollections;

import javafx.collections.ObservableList;

import javafx.event.ActionEvent;

import javafx.fxml.FXML;

import javafx.fxml.Initializable;

import javafx.scene.control.Alert;

import java.net.URL;

import java.sql.\*;

import java.util.ResourceBundle;

public class AddController implements Initializable {

@FXML

private JFXTextField dish\_name;

@FXML

private JFXButton add\_btn;

@FXML

private JFXComboBox<String> dish\_cat;

@FXML

private JFXTextField dish\_price;

ObservableList<String> list= FXCollections.observableArrayList();

PreparedStatement ps=null;

ResultSet rs=null;

@Override

public void initialize(URL location, ResourceBundle resources) {

try {

Connection con=DBConnection.getConnection();

String sql="select \* from categories";

ps=con.prepareStatement(sql);

rs=ps.executeQuery();

while(rs.next())

list.add(rs.getString("cat\_name"));

dish\_cat.setItems(list);

} catch (SQLException e) {

e.printStackTrace();

}

}

@FXML

public void add\_dish(ActionEvent actionEvent) throws SQLException {

Connection con=DBConnection.getConnection();

if(dish\_name.getText().isEmpty() || dish\_cat.getValue().isEmpty() || dish\_price.getText().isEmpty())

{

Alert errorAlert = new Alert(Alert.AlertType.ERROR);

errorAlert.setHeaderText("Input Error!");

errorAlert.setContentText("Please Enter Dish Name,Category and Price");

errorAlert.showAndWait();

}

else {

String check = "select \* from dishes where dish\_name='" + dish\_name.getText() + "'";

ps = con.prepareStatement(check);

rs = ps.executeQuery();

if (rs.next()) {

Alert errorAlert = new Alert(Alert.AlertType.ERROR);

errorAlert.setHeaderText("Data Error!");

errorAlert.setContentText("Dish Already Exists");

errorAlert.showAndWait();

} else {

String sql1="insert into dishes(dish\_name,cat\_id,cat\_name) values('"+dish\_name.getText()+"',(select cat\_id from categories where cat\_name='"+dish\_cat.getValue()+"'),'"+dish\_cat.getValue()+"')";

String sql2="insert into prices values((select dish\_id from dishes where dish\_name='"+dish\_name.getText()+"'),"+dish\_price.getText()+")";

ps = con.prepareStatement(sql1);

ps.executeUpdate();

ps = con.prepareStatement(sql2);

ps.executeUpdate();

Alert infoAlert = new Alert(Alert.AlertType.INFORMATION);

infoAlert.setHeaderText("Alert!");

infoAlert.setContentText("Dish Added Successfully!");

infoAlert.showAndWait();

dish\_name.setText("");

dish\_cat.setValue("");

dish\_price.setText("");

}

}

}

}

**4.2.4 Deleting Dishes:**

package sample;

import com.jfoenix.controls.JFXButton;

import com.jfoenix.controls.JFXTextField;

import javafx.event.ActionEvent;

import javafx.fxml.FXML;

import javafx.scene.control.Alert;

import javafx.scene.control.Label;

import java.sql.\*;

public class DeleteController {

@FXML

private JFXTextField dish\_id;

@FXML

private JFXButton search\_dish;

@FXML

private JFXButton del\_btn;

@FXML

private Label dish\_name;

@FXML

private Label dish\_cat;

@FXML

private Label dish\_price;

ResultSet rs=null;

PreparedStatement ps=null;

Connection con=DBConnection.getConnection();

public DeleteController() throws SQLException {

}

@FXML

public void delete\_dish(ActionEvent actionEvent) throws SQLException {

if(dish\_id.getText().isEmpty())

{

Alert errorAlert = new Alert(Alert.AlertType.ERROR);

errorAlert.setHeaderText("Input Error!");

errorAlert.setContentText("Please Enter Dish ID");

errorAlert.showAndWait();

}

else {

String check = "select \* from dishes where dish\_id='" + dish\_id.getText() + "'";

ps = con.prepareStatement(check);

rs = ps.executeQuery();

if (rs.next()) {

String sql = "delete from dishes where dish\_id=" + dish\_id.getText();

ps = con.prepareStatement(sql);

ps.executeUpdate();

Alert infoAlert = new Alert(Alert.AlertType.INFORMATION);

infoAlert.setHeaderText("Alert!");

infoAlert.setContentText("Dish Deleted Successfully!");

infoAlert.showAndWait();

dish\_id.setText("");

dish\_name.setText("");

dish\_cat.setText("");

dish\_price.setText("");

} else {

Alert errorAlert = new Alert(Alert.AlertType.ERROR);

errorAlert.setHeaderText("Data Error!");

errorAlert.setContentText("Invalid Dish ID!");

errorAlert.showAndWait();

}

}

}

@FXML

public void retrieve(ActionEvent actionEvent) throws SQLException

{

String sql1="select \* from dishes where dish\_id="+dish\_id.getText();

ps=con.prepareStatement(sql1);

rs=ps.executeQuery();

while(rs.next()) {

dish\_name.setText(rs.getString("dish\_name"));

dish\_cat.setText(rs.getString("cat\_name"));

}

String sql2 = "select \* from prices where dish\_id=" + dish\_id.getText();

ps = con.prepareStatement(sql2);

rs = ps.executeQuery();

while (rs.next()) {

dish\_price.setText(rs.getString("price"));

}

}

}

**4.2.5 Updating Dishes:**

package sample;

import com.jfoenix.controls.JFXButton;

import com.jfoenix.controls.JFXComboBox;

import com.jfoenix.controls.JFXTextField;

import javafx.collections.FXCollections;

import javafx.collections.ObservableList;

import javafx.event.ActionEvent;

import javafx.fxml.FXML;

import javafx.fxml.Initializable;

import javafx.scene.control.Alert;

import java.net.URL;

import java.sql.\*;

import java.util.ResourceBundle;

public class UpdateController implements Initializable {

@FXML

private JFXTextField dish\_id;

@FXML

private JFXButton search\_dish;

@FXML

private JFXTextField dish\_name;

@FXML

private JFXTextField dish\_price;

@FXML

private JFXComboBox<String> dish\_cat;

PreparedStatement ps=null;

ResultSet rs=null;

Connection con=DBConnection.getConnection();

ObservableList<String> list= FXCollections.observableArrayList();

public UpdateController() throws SQLException {

}

@FXML

public void update\_dish(ActionEvent actionEvent) throws SQLException {

if(dish\_id.getText().isEmpty() || dish\_name.getText().isEmpty() || dish\_cat.getValue().isEmpty() || dish\_price.getText().isEmpty())

{

Alert errorAlert = new Alert(Alert.AlertType.ERROR);

errorAlert.setHeaderText("Input Error!");

errorAlert.setContentText("Please Enter Product ID,Name,Category and Price");

errorAlert.showAndWait();

}

else {

String sql1="update dishes set dish\_name='"+dish\_name.getText()+"',cat\_id=(select cat\_id from categories where cat\_name='"+dish\_cat.getValue()+"'),cat\_name='"+dish\_cat.getValue()+"' where dish\_id="+dish\_id.getText();

ps=con.prepareStatement(sql1);

ps.executeUpdate();

String sql2="update prices set dish\_id="+dish\_id.getText()+",price="+dish\_price.getText()+" where dish\_id="+dish\_id.getText();

ps=con.prepareStatement(sql2);

ps.executeUpdate();

Alert errorAlert = new Alert(Alert.AlertType.ERROR);

errorAlert.setHeaderText("Alert!");

errorAlert.setContentText("Dish Updated Successfully!");

errorAlert.showAndWait();

dish\_name.setText("");

dish\_cat.setValue("");

dish\_price.setText("");

dish\_id.setText("");

}

}

@FXML

public void retrieve(ActionEvent actionEvent) throws SQLException {

String sql1="select \* from dishes where dish\_id="+dish\_id.getText();

ps=con.prepareStatement(sql1);

rs=ps.executeQuery();

while(rs.next())

{

dish\_name.setText(rs.getString("dish\_name"));

dish\_cat.getSelectionModel().select(rs.getString("cat\_name"));

}

String sql2="select \* from prices where dish\_id="+dish\_id.getText();

ps=con.prepareStatement(sql2);

rs=ps.executeQuery();

while(rs.next())

dish\_price.setText(rs.getString("price"));

}

@Override

public void initialize(URL location, ResourceBundle resources) {

String sql="select \* from categories";

try {

ps=con.prepareStatement(sql);

rs=ps.executeQuery();

while(rs.next())

list.add(rs.getString("cat\_name"));

dish\_cat.setItems(list);

} catch (SQLException e) {

e.printStackTrace();

}

}

}

**4.2.6 Displaying Dishes:**

package sample;

import com.jfoenix.controls.JFXTextField;

import javafx.collections.FXCollections;

import javafx.collections.ObservableList;

import javafx.collections.transformation.FilteredList;

import javafx.collections.transformation.SortedList;

import javafx.fxml.FXML;

import javafx.fxml.Initializable;

import javafx.scene.control.TableColumn;

import javafx.scene.control.TableView;

import javafx.scene.control.cell.PropertyValueFactory;

import java.net.URL;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ResourceBundle;

public class DisplayDishController implements Initializable {

@FXML

private TableView<ModelDishTable> dish\_table;

@FXML

private TableColumn<?, ?> dishid\_col;

@FXML

private TableColumn<?, ?> dishname\_col;

@FXML

private TableColumn<?, ?> catname\_col;

@FXML

private TableColumn<?, ?> catid\_col;

@FXML

private JFXTextField search;

ObservableList<ModelDishTable> oblist= FXCollections.observableArrayList();

@Override

public void initialize(URL location, ResourceBundle resources) {

try {

Connection con= DBConnection.getConnection();

ResultSet rs=con.createStatement().executeQuery("select \* from dishes");

while(rs.next())

{

oblist.add(new ModelDishTable(rs.getString("dish\_id"),rs.getString("dish\_name"),rs.getString("cat\_name"),rs.getString("cat\_id")));

}

} catch (SQLException e) {

e.printStackTrace();

}

dishid\_col.setCellValueFactory(new PropertyValueFactory<>("dish\_id"));

dishname\_col.setCellValueFactory(new PropertyValueFactory<>("dish\_name"));

catname\_col.setCellValueFactory(new PropertyValueFactory<>("cat\_name"));

catid\_col.setCellValueFactory(new PropertyValueFactory<>("cat\_id"));

dish\_table.setItems(oblist);

FilteredList<ModelDishTable> filteredData = new FilteredList<>(oblist, p -> true);

search.textProperty().addListener((observable, oldValue, newValue) -> {

filteredData.setPredicate(temp -> {

if (newValue == null || newValue.isEmpty()) {

return true;

}

String lowerCaseFilter = newValue.toLowerCase();

if (temp.getDish\_name().toLowerCase().contains(lowerCaseFilter)) {

return true;

}

return false;

});

});

SortedList<ModelDishTable> sortedData = new SortedList<>(filteredData);

sortedData.comparatorProperty().bind(dish\_table.comparatorProperty());

dish\_table.setItems(sortedData);

}

}

**4.2.4 Main Page:**

<html>

<head>

<title>Cricket Database</title>

<style>

ul li{

list-style:none;

}

a:link{

color:blue;

}

a:visited{

color:blue;

}

a:hover{

color:orange;

}

a:focus{

color:green;

}

a:active{

color:red;

}

</style>

</head>

<body style="margin:0px;padding:0px;">

<div id="wrapper" style="width:100%;height:100%;">

<div id="header" style="width:100%; height:20%;float:left;background-image: url('top.jpg');" >

<a href="index.html" style="float:right;">SignOut </a>

<center><h1 style="color:rgb(10, 235, 111);font-size:40px;"><br>Cricket Database</h1></center>

</div>

<div id="container" style=" width:39%; height:70%; float:left;background-image: url('bk2.jpg'); ;text-align:center;">

<h2>About</h2>

<p>This project displays the database </p>

<h3>Select Table To Retrieve</h3>

<ul>

<li><a href="coach.html" target="target">Coach</a></li>

<li><a href="players.html" target="target">Players</a></li>

<li><a href="team.html" target="target">Team</a></li>

<li><a href="plays.html" target="target">Plays</a></li>

<li><a href="matches.html" target="target">Matches</a></li>

<li><a href="scoreboard.html" target="target">Scoreboard</a></li>

<li><a href="umpire.html" target="target">Umpire</a></li>

<li><a href="umpiring.html" target="target">Umpiring</a></li>

</ul>

<!--<h3>To Retrieve Selected Columns OR To Execute Join and Various operations <a href="global.html" target="target">Click here</a></h3>

--></div>

<div id="target" style="width:60%;height:70%; float:left; background-image: url('bat.jpg');">

<iframe name="target" src="default2.html" style="width:100%;height:100%;"></iframe>

</div>

<div style="width:100%;height:10%;">

<img src="footer.jpg" style="width:100%;height:100%;">

</div> </div>

</body> </html>