import com.twilio.Twilio;

import com.twilio.rest.api.v2010.account.Message;

import com.twilio.type.PhoneNumber;

public class TwilioService {

private static final String ACCOUNT\_SID = "your\_account\_sid";

private static final String AUTH\_TOKEN = "your\_auth\_token";

private static final String TWILIO\_PHONE\_NUMBER = "your\_twilio\_phone\_number";

static {

Twilio.init(ACCOUNT\_SID, AUTH\_TOKEN);

}

public void sendOtp(String phoneNumber, String otp) {

Message.creator(new PhoneNumber(phoneNumber), new PhoneNumber(TWILIO\_PHONE\_NUMBER), "Your OTP is: " + otp).create();

}

import java.sql.\*;

public class DatabaseService {

private static final String DB\_URL = "jdbc:mysql://localhost:3306/otp\_db";

private static final String DB\_USER = "your\_db\_user";

private static final String DB\_PASSWORD = "your\_db\_password";

public void saveOtp(String phoneNumber, String otp) {

try (Connection connection = DriverManager.getConnection(DB\_URL, DB\_USER, DB\_PASSWORD)) {

String query = "INSERT INTO otps (phone\_number, otp) VALUES (?, ?)";

try (PreparedStatement statement = connection.prepareStatement(query)) {

statement.setString(1, phoneNumber);

statement.setString(2, otp);

statement.executeUpdate();

}

} catch (SQLException e) {

e.printStackTrace();

}

}

public boolean validateOtp(String phoneNumber, String otp) {

try (Connection connection = DriverManager.getConnection(DB\_URL, DB\_USER, DB\_PASSWORD)) {

String query = "SELECT \* FROM otps WHERE phone\_number = ? AND otp = ?";

try (PreparedStatement statement = connection.prepareStatement(query)) {

statement.setString(1, phoneNumber);

statement.setString(2, otp);

try (ResultSet resultSet = statement.executeQuery()) {

if (resultSet.next()) {

deleteOtp(phoneNumber, otp);

return true;

}

}

}

} catch (SQLException e) {

e.printStackTrace();

}

return false;

}

private void deleteOtp(String phoneNumber, String otp) {

try (Connection connection = DriverManager.getConnection(DB\_URL, DB\_USER, DB\_PASSWORD)) {

String query = "DELETE FROM otps WHERE phone\_number = ? AND otp = ?";

try (PreparedStatement statement = connection.prepareStatement(query)) {

statement.setString(1, phoneNumber);

statement.setString(2, otp);

statement.executeUpdate();

}

} catch (SQLException e) {

e.printStackTrace();

}

}

}

import javax.ws.rs.\*;

import javax.ws.rs.core.MediaType;

import javax.ws.rs.core.Response;

@Path("/otp")

public class OtpResource {

private TwilioService twilioService = new TwilioService();

private DatabaseService databaseService = new DatabaseService();

@POST

@Path("/send")

@Consumes(MediaType.APPLICATION\_JSON)

@Produces(MediaType.APPLICATION\_JSON)

public Response sendOtp(OtpRequest request) {

String otp = String.valueOf((int)(Math.random() \* 900000) + 100000);

twilioService.sendOtp(request.getPhoneNumber(), otp);

databaseService.saveOtp(request.getPhoneNumber(), otp);

return Response.ok().build();

}

@POST

@Path("/verify")

@Consumes(MediaType.APPLICATION\_JSON)

@Produces(MediaType.APPLICATION\_JSON)

public Response verifyOtp(OtpRequest request) {

boolean isValid = databaseService.validateOtp(request.getPhoneNumber(), request.getOtp());

if (isValid) {

return Response.ok().build();

} else {

return Response.status(Response.Status.UNAUTHORIZED).build();

}

}

public static class OtpRequest {

private String phoneNumber;

private String otp;

public String getPhoneNumber() {

return phoneNumber;

}

public void setPhoneNumber(String phoneNumber) {

this.phoneNumber = phoneNumber;

}

public String getOtp() {

return otp;

}

public void setOtp(String otp) {

this.otp = otp;

}

}

}