

Technical Document

PAT Phase 4 – Grade 12

Prepared by Marco de Sá

Maragon Mooikloof

10/6/2022

Table of Contents

1. Externally Sourced Code 1

2. Explanation of Critical Algorithms 1

3. Advanced Techniques 2

# Externally Sourced Code

[This section is intended for you to declare any code that you used that you did not write directly. This includes any of the following:

• Database connection strings

• Database classes that were given out by your teacher

• Libraries for emailing or PDF creation

• "Cool" code that you obtained from other learners in your class (e.g. opening an external program, login screens/code, visually appealing graphics or music-playing code.

-DB Class

public void deleteEmployeeFromDB(int ID) {

changeDB("DELETE FROM tblEmployees WHERE EmployeeID = " + ID);

}

private void changeDB(String inStatement) {

try {

HobbyZA.changeDB(inStatement);

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, "could not alter");

}

}

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

if (txtUsername.getText().equals("")) {

JOptionPane.showMessageDialog(null, "No username entered");

}else if(pwfPassword.getPassword().length == 0){

JOptionPane.showMessageDialog(null, "No Password entered");

}else if(!users.passwordValid(txtUsername.getText(), pwfPassword)){

JOptionPane.showMessageDialog(null, "Incorrect Password");

}else{

new FrmMenu().setVisible(true);

this.dispose();

}

}

public boolean userExists(String inUserName){

boolean found = false;

for (int i = 0; i < size; i++) {

if (userArr[i].getUserName().equals(inUserName)) {

found = true;

break;

}

}

return found;

}

# Explanation of Critical Algorithms

Login searches if a password and username is inputed and then checks if it is a valid username and password

populateEmployeeData() refreshes the text area by reading the updated data from the database and then updates the text area with the new data

updateEmployeeData() takes data and puts it into a query that updates the database

addEmployeeToDB() takes data and puts it into a query that adds data to the database

deleteEmployeeFromDB() deletes the data associated with a certain employee ID

getEmployeeData() returns all data in the database

# Advanced Techniques

There are no especially advanced techniques