**Experiment Report - 13 – test4\_Page12**

1. **Summary Table of Errors Found**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Error ID | Line Number | Error Type | CSCR? | Self-Detected? | LLM? |
| E01 | Line 21 | Semantic | √ |  | × |
| E02 | Line 33 | Syntax | × |  | × |
| E03 | Line 62 | Logic | √ |  | × |
| E04 | Line 55 (74) | Syntax |  | √ | √ |
| E05 | Line 74 (93) | Semantic |  | √ | √ |
| E06 | Line 92 (111) | Semantic |  | × | √ |

Additional Errors Found by Self: 0

CSCR Rate: 67%

Self-Review Detection Rate: 67%

LLM Rate: 50%

1. **Source Code**
2. import javax.swing.\*;
3. import java.awt.\*;
4. import java.awt.event.ActionEvent;
5. import java.awt.event.ActionListener;
6. import java.io.BufferedReader;
7. import java.io.FileReader;
8. import java.io.IOException;
9. import java.util.ArrayList;
10. import java.util.Arrays;
11. import java.util.List;
12. public class Page12 extends JFrame {
13. public static void main(String args[]) {
14. Page12 frame = new Page12("Page12");
15. frame.setVisible(true);
16. }
17. String cop = "udata.csv";
18. String line = "";
19. String Split = ",";
20. List<List<String>> data = new ArrayList<>();
21. private String copName;
22. private String copPass;
23. Page12(String title) {
24. setTitle(title);
25. setSize(1500, 900);
26. setDefaultCloseOperation(EXIT\_ON\_CLOSE);
27. JPanel p1 = new JPanel();
28. JPanel p2 = new JPanel();
29. JPanel p3 = new JPanel();
30. JTextField name = new JTextField(15);
31. JLabel label1 = new JLabel("掲載誌");
32. JPasswordField pass = new JPasswordField(15);
33. JLabel label2 = new JLabel("パスワード");
34. JButton check = new JButton("確認");
35. check.addActionListener(new ActionListener() {
36. public void actionPerformed(ActionEvent e) {
37. try (BufferedReader br = new BufferedReader(new FileReader(cop))) {
38. copName = name.getText();
39. char[] password = pass.getPassword();
40. copPass = String.valueOf(password);
41. while ((line = br.readLine()) != null) {
42. *// CSVファイルをカンマで分割*
43. String[] values = line.split(Split);
44. *// 配列をリストに変換して追加*
45. data.add(Arrays.asList(values));
46. }
47. boolean found = false;
48. for (List<String> record : data) {
49. if (record.size() > 3) { *// 行に十分なデータがあることを確認*
50. String name = record.get(0);
51. String pass = record.get(1);
52. *// 比較*
53. if (name.equals(copName) && pass.equals(copPass)) {
54. found = true;
55. break;
56. }
57. }
58. }
59. System.out.println(copName + copPass);
60. if (found = true) {
61. Page14 frame = new Page14(title, copName, copPass);
62. frame.setVisible(true);
63. dispose();
64. }
65. else{
66. JOptionPane.showMessageDialog(null, "掲載誌とパスワードが一致しません。");
67. }
68. } catch (IOException ee) {
69. ee.printStackTrace();
70. }
71. }
72. });
73. ImageIcon origin = new ImageIcon("./bar.png");
74. Image original = origin.getImage();
75. Image resize = original.getScaledInstance(100, 100, Image.SCALE\_SMOOTH);
76. ImageIcon resizedback = new ImageIcon(resize);
77. JButton back = new JButton(resized);
78. back.setPreferredSize(new Dimension(resizedback.getIconWidth(), resizedback.getIconHeight()));
79. back.addActionListener(new ActionListener() {
80. public void actionPerformed(ActionEvent e) {
81. Page11 frame = new Page11(title);
82. frame.setVisible(true);
83. dispose();
84. }
85. });
86. p1.add(name);
87. p1.add(label1);
88. p1.add(pass);
89. p1.add(label2);
90. p2.add(back);
91. p3.add(back);
92. Container contentPane = getContentPane();
93. contentPane.add(p1, BorderLayout.CENTER);
94. contentPane.add(p2, BorderLayout.WEST);
95. contentPane.add(p3, BorderLayout.EAST);
96. }
97. }