**Experiment Report - 01 - test1\_page2**

1. **Summary Table of Errors Found**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Error ID | Line Number | Error Type | CSCR? | Self-Detected? | LLM? |
| E01 | Line 23 | Semantic | √ |  | × |
| E02 | Line 32 | Semantic | √ |  | √ |
| E03 | Line 42 | Semantic | √ |  | × |
| E04 | Line 50 | Syntax | √ |  | √ |
| E05 | Line 61 | Logic | √ |  | × |
| E06 | Line 35 (85) | Logic |  | × | √ |
| E07 | Line 42 (92) | Semantic |  | √ | × |
| E08 | Line 50 (100) | Semantic |  | √ | √ |
| E09 | Line 56 (106) | Syntax |  | × | × |
| E10 | Line 62 (112) | Syntax |  | × | √ |

Additional Errors Found by Self: 1

CSCR Rate: 100%

Self-Review Detection Rate: 40%

LLM Rate: 50%

1. **Source Code**
2. import javax.swing.\*;
3. import java.awt.\*;
4. import java.awt.event.\*;
5. import java.time.LocalDateTime;
6. public class page2 extends JPanel {
7. private JTextField amountTextField;
8. private JRadioButton pointUseButton, pointNotUseButton;
9. private ButtonGroup buttonGroup;
10. private JButton paymentButton;
11. private boolean usePoint;
12. public static final Integer[] COUPON\_RATES = {0, 10, 20, 30, 40, 50};
13. private JComboBox<Integer> couponComboBox;
14. private JLabel discountedAmountLabel;
15. public page2() {
16. *// レイアウトの設定*
17. setLayout(new FlowLayout()); *// レイアウトをFlowLayoutに変更*
19. *// 支払金額入力欄*
20. JLabel amountLabel = new JLabel("支払金額:");
21. add(amountLabel);
22. amountTextField = new JTextField(5); *// 幅を指定*
23. add(amountTextField);
24. add(new JLabel("円"));
25. *// ポイント使用ラジオボタン*
26. pointUseButton = new JRadioButton("ポイントを使用する");
27. buttonGroup = new ButtonGroup();
28. buttonGroup.add(pointUseButton);
29. pointNotUseButton = new JRadioButton("使用しない");
30. buttonGroup.add(pointUseButton);
31. JPanel radioPanel = new JPanel();
32. radioPanel.add(pointUseButton);
33. radioPanel.add(pointNotUseButton);
34. add(radioPanel);
35. *// クーポン選択*
36. JLabel couponLabel = new JLabel("クーポンを選ぶ");
37. add(couponLabel);
38. couponComboBox = new JComboBox<>(COUPON\_RATES);
39. couponComboBox.setSelectedIndex(1);
40. add(couponComboBox);
41. add(new JLabel("%"));
42. *// 決済ボタン*
43. paymentButton = new JButton("決済");
44. paymentButton.addActionListener(new ActionListener() {
45. @Override
46. public void actionPerformed(ActionEvent) {
47. int paymentAmount = Integer.parseInt(amountTextField.getText());
48. usePoint = pointUseButton.isSelected();
49. *// クーポン割引率を取得*
50. int couponRate = (int) couponComboBox.getSelectedItem();
51. *// 決済金額を計算 (クーポン割引を適用)*
52. int discountedAmount = paymentAmount - (paymentAmount \* couponRate / 100);
53. *// 残高チェック (割引後の金額でチェック)*
54. if (discountedAmount >= UserData.getBalance()) {
55. JOptionPane.showMessageDialog(null, "残高が不足しています。");
56. return;
57. }
59. *// 割引後の金額を表示*
60. discountedAmountLabel.setText("割引後金額: " + discountedAmount + "円");
61. }
62. });
63. add(paymentButton);
64. *// 割引後の金額を表示するラベル*
65. discountedAmountLabel = new JLabel();
66. add(discountedAmountLabel);
67. }
68. *// 決済ボタン*
69. paymentButton = new JButton("決済");
70. paymentButton.addActionListener(new ActionListener() {
71. @Override
72. public void actionPerformed(ActionEvent e) {
73. *// ポイント使用時の処理*
74. if (usePoint) {
75. int currentPoint = UserData.getPoint();
76. if (currentPoint > discountedAmount) {
77. *// ポイントで全額支払う*
78. UserData.setPoint(currentPoint - discountedAmount);
79. discountedAmount = 0;
80. } else {
81. *// ポイントで支払えるだけ支払う*
82. discountedAmount -= currentPoint;
83. UserData.setPoint(currentPoint);
84. }
85. }
86. *// 残高の更新*
87. UserData.setBalance(UserData.getBalance() - discountedAmount);
88. *// ポイントの付与 (100円ごとに1ポイント)*
89. int earnedPoint = discountedAmount / 10;
90. UserData.setPoint(UserData.getPoint() + earnedPoint);
91. int usedPoint = usePoint ? Math.min(discountedAmount, UserData.getPoint()) : 0;
92. *// 決済履歴の保存*
93. LocalDateTime now = LocalDateTime.now();
94. PaymentLog.save(discountedAmount, usedPoint, now);
95. *// 決済完了メッセージを表示*
96. JOptionPane.showMessageDialog(null, "決済が完了しました。");
97. *// MainFrameのインスタンスを取得して、updatePages()メソッドを呼び出す*
98. MainFrame mainFrame = (MainFrame) SwingUtilities.getwindowAncestor(page2.this);
99. mainFrame.updatePages();
101. *// 割引後の金額を表示*
102. discountedAmountLabel.setText("割引後金額: " + discountedAmount + "円");
103. }
104. });
105. add(paymentButton);
106. *// 割引後の金額を表示するラベル*
107. discountedAmountLabel = new JLabel();
108. add(discountedAmountLabel);
109. }
110. }