**Experiment Report - 02 - test2\_page2**

1. **Summary Table of Errors Found**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Error ID | Line Number | Error Type | Self-Detected? | Peer 1 Found? | Peer 2 Found? |
| E01 | Line 35 | Logic | √ | × | × |
| E02 | Line 42 | Semantic | × | × | × |
| E03 | Line 50 | Semantic | √ | √ | √ |
| E04 | Line 56 | Syntax | √ | × | × |
| E05 | Line 62 | Syntax | × | × | × |

Additional Errors Found by Self: 0

Self-Review Detection Rate: 60%

Peer 1 Detection Rate: 20%

Peer 2 Detection Rate: 20%

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(10); // 幅を指定
23. add(amountTextField);
24. add(new JLabel("円"));
25. // 決済ボタン
26. paymentButton = new JButton("決済");
27. paymentButton.addActionListener(new ActionListener() {
28. @Override
29. public void actionPerformed(ActionEvent e) {
30. // ポイント使用時の処理
31. if (usePoint) {
32. int currentPoint = UserData.getPoint();
33. if (currentPoint > discountedAmount) {
34. // ポイントで全額支払う
35. UserData.setPoint(currentPoint - discountedAmount);
36. discountedAmount = 0;
37. } else {
38. // ポイントで支払えるだけ支払う
39. discountedAmount -= currentPoint;
40. UserData.setPoint(currentPoint);
41. }
42. }
43. // 残高の更新
44. UserData.setBalance(UserData.getBalance() - discountedAmount);
45. // ポイントの付与 (100円ごとに1ポイント)
46. int earnedPoint = discountedAmount / 10;
47. UserData.setPoint(UserData.getPoint() + earnedPoint);
48. int usedPoint = usePoint ? Math.min(discountedAmount, UserData.getPoint()) : 0;
49. // 決済履歴の保存
50. LocalDateTime now = LocalDateTime.now();
51. PaymentLog.save(discountedAmount, usedPoint, now);
52. // 決済完了メッセージを表示
53. JOptionPane.showMessageDialog(null, "決済が完了しました。");
54. // MainFrameのインスタンスを取得して、updatePages()メソッドを呼び出す
55. MainFrame mainFrame = (MainFrame) SwingUtilities.getwindowAncestor(page2.this);
56. mainFrame.updatePages();
58. // 割引後の金額を表示
59. discountedAmountLabel.setText("割引後金額: " + discountedAmount + "円");
60. }
61. });
62. add(paymentButton);
63. // 割引後の金額を表示するラベル
64. discountedAmountLabel = new JLabel();
65. add(discountedAmountLabel);
66. }
67. }