

```

1 package view;
2
3 import java.awt.Color;
23
24 /**
25  * Class LogInInterface to construct interface presented after successful login.
26  * @author Daria Vekic (Student ID: 586661)
27  *
28  */
29 public class LogInInterface extends JFrame {
30
31     private static final long serialVersionUID = 1L;
32     private JPanel contentPane;
33     private JTextField usernameTextField;
34     private JPasswordField passwordField;
35     private JLabel resetLink;
36     private JButton signInButton;
37     private LogInController controller = new LogInController();
38     private CommonElements common = new CommonElements();
39
40     /**
41      * Constructor method to create LogInInterface object.
42      * @param employees the List of Employees
43      * @param clients the List of Clients
44      * @param cases the List of Cases
45      */
46     public LogInInterface(ArrayList<Employee> employees, ArrayList<Client> clients,
47         ArrayList<Case> cases) {
48         createLogInInterface(employees, clients, cases);
49     } //end constructor method
50
51     /**
52      * Method to create the GUI to allow a user to log in to system.
53      * @param employees the List of Employees
54      * @param clients the List of Clients
55      * @param cases the List of Cases
56      */
57     private void createLogInInterface(ArrayList<Employee> employees, ArrayList<Client> clients,
58         ArrayList<Case> cases) {
59         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
60         setBounds(100, 100, 622, 644);
61         contentPane = new JPanel();
62         contentPane.setBackground(Color.WHITE);
63         setContentPane(contentPane);
64         contentPane.setLayout(null);
65
66         JLabel imgLabel = new JLabel("");
67         imgLabel.setIcon(new ImageIcon(LogInInterface.class.getResource("/img/Logo.png")));
68         imgLabel.setBounds(142, 11, 300, 300);
69         contentPane.add(imgLabel);
70
71         usernameTextField = new JTextField(30);
72         usernameTextField.addMouseListener(new MouseAdapter() {
73             @Override

```

```

74         public void mouseClicked(MouseEvent e) {
75             usernameTextField.setText("");
76         }
77     });
78     usernameTextField.setToolTipText("Enter your username here");
79     usernameTextField.setFont(new Font("Verdana", Font.PLAIN, 14));
80     usernameTextField.setText("Username");
81     usernameTextField.setBounds(152, 325, 283, 37);
82     contentPane.add(usernameTextField);
83     usernameTextField.setColumns(10);
84
85     passwordField = new JPasswordField(25);
86     passwordField.addMouseListener(new MouseAdapter() {
87         @Override
88         public void mouseClicked(MouseEvent e) {
89             passwordField.setText("");
90         }
91     });
92     passwordField.setFont(new Font("Verdana", Font.PLAIN, 14));
93     passwordField.setToolTipText("Enter your password here");
94     //passwordField.setText("Password");
95     passwordField.setBounds(152, 373, 283, 37);
96     contentPane.add(passwordField);
97     passwordField.setColumns(10);
98
99     resetLink = new JLabel("Click here to set or change your password");
100    resetLink.setFont(new Font("Verdana", Font.PLAIN, 12));
101    resetLink.setBounds(155, 421, 300, 21);
102    resetLink.setForeground(Color.blue.darker());
103    resetLink.setCursor(Cursor.getPredefinedCursor(Cursor.HAND_CURSOR));
104    resetLink.addMouseListener(new MouseAdapter() {
105        @Override
106        public void mouseClicked(MouseEvent e) {
107            resetLinkClicked(e, employees, cases, clients);
108        }
109    });
110    contentPane.add(resetLink);
111
112    signInButton = new JButton("Sign In");
113    common.styleButton(signInButton);
114    signInButton.setBounds(211, 466, 154, 37);
115    signInButton.addActionListener(new ActionListener()
116    {
117        public void actionPerformed(ActionEvent event) {
118            signInButtonActionPerformed(event, employees, cases, clients);
119        } //end method actionPerformed
120    } //end method ActionListener
121    ); //end call to signInButtonActionPerformed ActionListener
122    contentPane.add(signInButton);
123
124    // set properties of application's window
125    setTitle("McRae & Dick Log In"); // set title bar string
126    setLocationRelativeTo(null);
127    setVisible(true);

```

```
128     } //end method createLogInInterface
129
130     /**
131     * Routine to be followed when link to reset password is clicked.
132     * @param event the "Reset Password" link is clicked
133     * @param employees the List of Employees
134     * @param cases the List of Cases
135     * @param clients the List of Clients
136     */
137     private void resetLinkClicked(MouseEvent event, ArrayList<Employee> employees,
138         ArrayList<Case> cases, ArrayList<Client> clients) {
139         //if the user clicks the link we want to display the ResetPassword window
140         this.setVisible(false);
141         ResetPasswordInterface rp = new ResetPasswordInterface(employees, cases, clients);
142         rp.setVisible(true);
143     } //end method resetLinkClicked
144
145     /**
146     * Routine to be followed when "Sign In" button is pressed.
147     * @param event the "Sign In" button is pressed
148     * @param employees the List of Employees
149     * @param cases the List of Cases
150     * @param clients the List of Clients
151     */
152     private void signInButtonActionPerformed(ActionEvent event, ArrayList<Employee> employees,
153         ArrayList<Case> cases, ArrayList<Client> clients) {
154         if(controller.signIn(usernameTextField, passwordField)) {
155             this.dispose();
156             HomeInterface home = new HomeInterface(employees, clients, cases);
157             home.setVisible(true);
158         } //endif
159     } //end method signInButtonActionPerformed
160 } //end class LogInInterface
```