

Team 18 Report

Work Allocation:

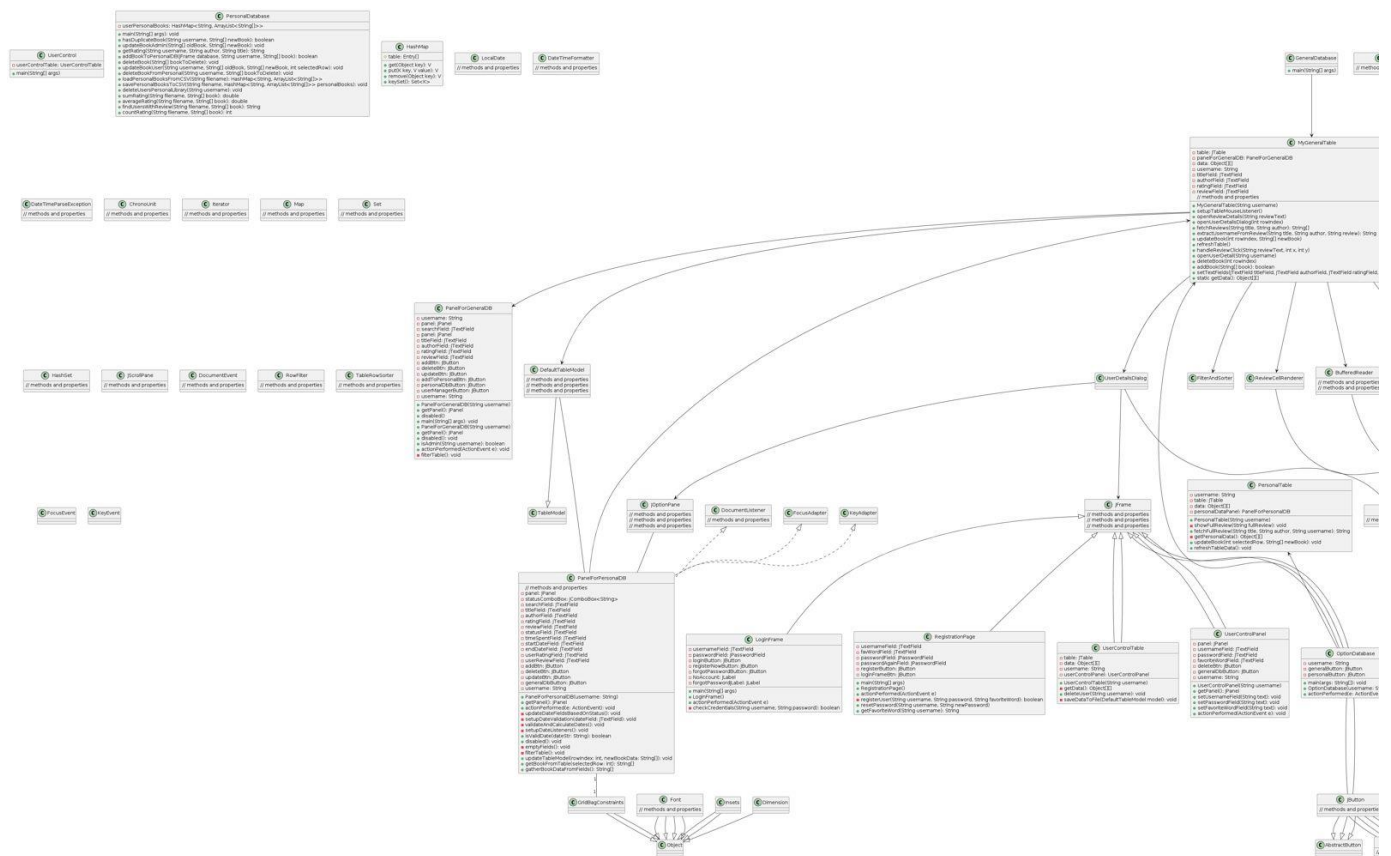
Full Name	Student ID	Approximate % of Work Done	What was Done
Omar Guluyev	17962	25%	Prepared the login screen and made sure you could log in by checking your username and password.
Murad Taghizade	18130	25%	Made the buttons for choosing what you want to do after logging in, like accessing your personal or general database, and made sure everything looked good on the screen.
Rustam Tahmazov	16277	25%	Focused on making the code feasible, readable, and conducive to further development.
Sadig Hasanli	18131	25%	Dedicated their efforts to ensuring that all project classes functioned as intended. By testing and debugging the code, resolved any issues that arose during development.

Our GitHub Link:

<https://github.com/ADA-SITE-CSCI-1202/team-project-team-18>

Our YouTube video link ; <https://youtu.be/-spMlaEZVXE>

UML Diagram for Our Project:



Code Snippets:

FilterAndSorter.java class:

```

import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.util.*;

public class FilterAndSorter {
    private static final int ASCENDING = 0;
    private static final int DESCENDING = 1;
    private static final int ORIGINAL = 2;

    private static Map<Integer, Integer> sortStates = new HashMap<>();
    private static ArrayList<Object[]> dataInitial;

    public static void sortSelected(JTable table) {
        initializeTable(table);
        table.getTableHeader().addMouseListener(new SortMouseListener(table));
    }

    private static void initializeTable(JTable table) {
        DefaultTableModel m = (DefaultTableModel) table.getModel();
        dataInitial = new ArrayList<>();
        for (int z = 0; z < m.getRowCount(); z++) {
            Object[] currentRow = new Object[m.getColumnCount()];
            for (int x = 0; x < m.getColumnCount(); x++) {
                currentRow[x] = m.getValueAt(z, x);
            }
            dataInitial.add(currentRow);
        }
    }

    static class SortMouseListener extends MouseAdapter {
        private final JTable table;

        SortMouseListener(JTable table) {
            this.table = table;
        }

        @Override
        public void mouseClicked(MouseEvent e) {
            int columnIndex = table.columnAtPoint(e.getPoint());
            if (columnIndex != -1) { // Check if a valid column was clicked
                int currentState = sortStates.getOrDefault(columnIndex, ORIGINAL);
                currentState = (currentState + 1) % 3; // Cycle through ASCENDING, DESCENDING, ORIGINAL

                // Update sort state for the column
                sortStates.put(columnIndex, currentState);

                DefaultTableModel currentModel = (DefaultTableModel) table.getModel();
                ArrayList<Object[]> data = new ArrayList<>(dataInitial);

                // Apply sorting based on all columns' sort states
                data.sort(new MultiColumnComparator(sortStates));

                // Update the table with sorted data
                currentModel.setRowCount(0);
                for (Object[] r : data) {
                    currentModel.addRow(r);
                }
            }
        }
    }

    static class MultiColumnComparator implements Comparator<Object[]> {
        private final Map<Integer, Integer> sortStates;

        MultiColumnComparator(Map<Integer, Integer> sortStates) {
            this.sortStates = sortStates;
        }

        @Override
        public int compare(Object[] o1, Object[] o2) {
            for (Map.Entry<Integer, Integer> entry : sortStates.entrySet()) {
                int columnIndex = entry.getKey();
                int sortOrder = entry.getValue();

                Comparable<Object> value1 = (Comparable<Object>) o1[columnIndex];
                Comparable<Object> value2 = (Comparable<Object>) o2[columnIndex];

                int result = value1.compareTo(value2);
                if (result != 0) {
                    return sortOrder == ASCENDING ? result : sortOrder == DESCENDING ? -result : 0;
                }
            }
            return 0; // Default: no difference
        }
    }
}

```

ForgotPassword.java class:

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class ForgotPassword extends JFrame implements ActionListener {
    private JButton renewPasswordButton;
    private JLabel favoriteWordJLabel;
    private JTextField usernameField, favoriteWordField;
    private JLabel passwordLabel, passwordAgainLabel;
    private JPasswordField passwordField, passwordAgainField;
    private JButton backToLoginButton;

    public ForgotPassword() {
        setTitle("Forgot Password");
        setLayout(new GridBagLayout());
        setSize(450, 400);
        GridBagConstraints gbc = new GridBagConstraints();
        gbc.gridx = 0;
        gbc.gridy = 0;
        gbc.anchor = GridBagConstraints.WEST;
        gbc.insets = new Insets(5, 5, 5, 5);

        JLabel usernameLabel = new JLabel("Username:");
        add(usernameLabel, gbc);

        gbc.gridy++;
        usernameField = new JTextField(15);
        add(usernameField, gbc);

        gbc.gridy++;
        passwordLabel = new JLabel("Password:");
        add(passwordLabel, gbc);

        gbc.gridy++;
        passwordField = new JPasswordField(15);
        add(passwordField, gbc);

        gbc.gridy++;
        passwordAgainLabel = new JLabel("Password Again:");
        add(passwordAgainLabel, gbc);

        gbc.gridy++;
        passwordAgainField = new JPasswordField(15);
        add(passwordAgainField, gbc);

        gbc.gridy++;
        favoriteWordJLabel = new JLabel("Write down your most favorite word");
        add(favoriteWordJLabel, gbc);

        gbc.gridy++;
        favoriteWordField = new JTextField(15);
        add(favoriteWordField, gbc);

        gbc.gridy++;
        renewPasswordButton = new JButton("Renew Password");
        renewPasswordButton.addActionListener(this);
        add(renewPasswordButton, gbc);

        gbc.gridy++;
        backToLoginButton = new JButton("Back to Login");
        backToLoginButton.addActionListener(this);
        gbc.gridwidth = 2;
        gbc.fill = GridBagConstraints.HORIZONTAL;
        add(backToLoginButton, gbc);
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == renewPasswordButton) {
            String username = usernameField.getText().toLowerCase().trim();
            String password = new String(passwordField.getPassword()).toLowerCase();
            String passwordAgain = new String(passwordAgainField.getPassword()).toLowerCase();
            String favoriteWord = favoriteWordField.getText().toLowerCase().trim();

            if (checkFavoriteWord(username, favoriteWord)) {
                if (password.trim().equals(passwordAgain.trim())) {
                    resetPassword(username, password);
                    JOptionPane.showMessageDialog(this, "Password reset successful.");
                } else {
                    JOptionPane.showMessageDialog(this, "Make sure that password matches to the password repetition.");
                }
            } else {
                JOptionPane.showMessageDialog(this, "Favorite word does not match.");
            }
        } else if (e.getSource() == backToLoginButton) {
            this.dispose();
            LoginFrame loginFrame = new LoginFrame();
            loginFrame.setLocationRelativeTo(null);
            loginFrame.setVisible(true);
        }
    }

    private boolean checkFavoriteWord(String username, String favoriteWord) {
        String storedFavoriteWord = RegistrationPage.getFavoriteWord(username);
        return favoriteWord.equals(storedFavoriteWord);
    }

    private void resetPassword(String username, String password) {
        RegistrationPage.resetPassword(username, password);
        JOptionPane.showMessageDialog(this, "Resetting password for username: " + username + ", New password: " + password);
    }
}

```

LoginFrame.java class:

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.*;

public class loginFrame extends JFrame implements ActionListener {
    private JTextField usernameField;
    private JPasswordField passwordField;
    private JButton loginButton;
    private JButton registerNowButton;
    private JButton forgotPasswordButton;
    private JLabel NoAccount;
    private JLabel forgotPasswordLabel;

    public static void main(String[] args) {
        loginFrame loginFrame = new loginFrame();
        loginFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        loginFrame.setVisible(true);
    }

    public loginFrame() {
        setTitle("Home Library login");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(450, 400);

        JPanel panel = new JPanel(new GridBagLayout());
        GridBagConstraints gbc = new GridBagConstraints();
        gbc.gridx = 0;
        gbc.gridy = 0;
        gbc.anchor = GridBagConstraints.CENTER;
        gbc.insets = new Insets(5, 5, 5, 5);

        JLabel label = new JLabel("Welcome Our Library");
        label.setFont(new Font("Arial", Font.BOLD, 14));
        panel.add(label, gbc);

        gbc.gridy++;
        JLabel usernameLabel = new JLabel("Username:");
        panel.add(usernameLabel, gbc);

        usernameField = new JTextField(15);
        gbc.gridy++;
        panel.add(usernameField, gbc);

        gbc.gridy++;
        JLabel passwordLabel = new JLabel("Password:");
        panel.add(passwordLabel, gbc);

        passwordField = new JPasswordField(15);
        gbc.gridy++;
        panel.add(passwordField, gbc);

        gbc.gridy++;
        loginButton = new JButton("Login");
        loginButton.addActionListener(this);
        panel.add(loginButton, gbc);

        NoAccount = new JLabel("Don't you have an account?");
        gbc.gridy++;
        panel.add(NoAccount, gbc);

        gbc.gridy++;
        registerNowButton = new JButton("Register");
        registerNowButton.addActionListener(this);
        panel.add(registerNowButton, gbc);

        gbc.gridy++;
        forgotPasswordLabel = new JLabel("Did you forgot your password?");
        panel.add(forgotPasswordLabel, gbc);

        gbc.gridy++;
        forgotPasswordButton = new JButton("Forgot Password");
        forgotPasswordButton.addActionListener(this);
        panel.add(forgotPasswordButton, gbc);

        add(panel);
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == loginButton) {
            String username = usernameField.getText().toLowerCase().trim();
            String password = new String(passwordField.getPassword()).toLowerCase();
            if (username.equals("admin") && password.equals("admin")) {
                new MyGenerator(username);
                PanelForGenerateID.disabled();
            }

            else if (checkCredentiaals(username, password)) {
                JOptionPane.showMessageDialog(this, "Login Successful");

                dispose();

                OptionDatabase optionDatabase = new OptionDatabase(username);
                optionDatabase.setVisible(true);
            }
            else {
                JOptionPane.showMessageDialog(this, "Invalid Username or Password");
            }
        }
        else if (e.getSource() == registerNowButton) {
            this.dispose();
            RegistrationPage rp = new RegistrationPage();
            rp.setLocationRelativeTo(null);
            rp.setVisible(true);
        }
        else if (e.getSource() == forgotPasswordButton) {
            this.dispose();
            ForgotPassword fp = new ForgotPassword();
            fp.setLocationRelativeTo(null);
            fp.setVisible(true);
        }
    }

    private boolean checkCredentiaals(String username, String password) {
        try {
            BufferedReader br = new BufferedReader(new InputStreamReader("users.csv"));
            String line;
            while ((line = br.readLine()) != null) {
                String[] parts = line.split(",");
                if (parts.length > 2 && parts[0].toLowerCase().equals(username) && parts[1].toLowerCase().equals(password)) {
                    return true;
                }
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
        return false;
    }
}

```

OptionDatabase.java class:

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class OptionDatabase extends JFrame implements ActionListener {
    String username;
    public static void main(String[] args) {
        //OptionDatabase optionDatabase = new OptionDatabase();
        // optionDatabase.setVisible(true);
    }

    JButton generalButton;
    JButton personalButton;

    OptionDatabase(String username){
        this.username=username;
        setTitle("Option Database");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(450, 200);
        setLayout(null);
        setLocationRelativeTo(null);
        setResizable(false);
        generalButton=new JButton("General DB");
        generalButton.setBounds(50, 100, 160, 25);
        generalButton.setFont(new Font("Arial", Font.BOLD, 14));

        generalButton.addActionListener(this);
        generalButton.setFocusPainted(false);
        add(generalButton);

        personalButton=new JButton("Personal DB");
        personalButton.setBounds(230, 100, 160, 25);
        personalButton.setFont(new Font("Arial", Font.BOLD, 14));
        personalButton.addActionListener(this);
        add(personalButton);

    }

    @Override
    public void actionPerformed(ActionEvent e){
        if (e.getSource() == generalButton){
            new MyGeneralTable(username);
        }
        else if (e.getSource() == personalButton)
        {
            new PersonalTable(username);
        }
    }
}

```

User.java class:

```
public class User {
    String username;
    private String password;

    public User(String username, String password) {
        this.username = username;
        this.password = password;
    }

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public void setUser(User user)
    {
        this.username=user.getUsername();
        this.password=user.getPassword();
    }
    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public boolean isAdmin(String username, String password)
    {
        return (username=="admin")&&(password=="admin");
    }
}
```


UserDetailsDialog.java class:

```
import javax.swing.*;
import java.awt.*;

public class UserDetailsDialog extends JDialog {
    public UserDetailsDialog(Frame owner, String username, String title, String author, String rating, String[] reviews) {
        super(owner, "Review Details", true);
        setDefaultCloseOperation(DISPOSE_ON_CLOSE);
        setSize(300, 200);
        setLayout(new BorderLayout());

        JTextArea textArea = new JTextArea();
        textArea.setEditable(false);
        for (String review : reviews) {
            textArea.append("Username: " + username + "\n" +
                "Title: " + title + "\n" +
                "Author: " + author + "\n" +
                "Rating: " + rating + "\n" +
                "Review: " + review + "\n\n");
        }
        JScrollPane scrollPane = new JScrollPane(textArea);
        add(scrollPane, BorderLayout.CENTER);

        JButton closeButton = new JButton("Close");
        closeButton.addActionListener(e -> dispose());
        add(closeButton, BorderLayout.SOUTH);

        setLocationRelativeTo(owner);
    }
}
```

UserControlPanel.java class:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class UserControlPanel extends JFrame implements ActionListener {
    private JPanel panel;
    private JTextField usernameField, passwordField, favoriteWordField;
    private JButton deleteBtn, generalDbButton;
    String username;

    public UserControlPanel(String username) {
        this.username = username;
        panel = new JPanel();
        panel.setLayout(null);
        usernameField = new JTextField();
        passwordField = new JTextField();
        favoriteWordField = new JTextField();
        deleteBtn = new JButton("Delete User");
        generalDbButton = new JButton("GeneralDB");

        JLabel crudLabel = new JLabel("User Manager");
        crudLabel.setFont(new Font("Arial", Font.BOLD, 22));
        crudLabel.setBounds(125, 160, 200, 40);
        panel.add(crudLabel);

        JLabel usernameLabel = new JLabel("Username: ");
        usernameLabel.setBounds(40, 250, 150, 30);
        usernameLabel.setFont(new Font("Arial", Font.BOLD, 14));
        panel.add(usernameLabel);
        usernameField.setBounds(160, 250, 200, 30);
        panel.add(usernameField);

        JLabel passwordLabel = new JLabel("Password: ");
        passwordLabel.setBounds(40, 300, 150, 30);
        passwordLabel.setFont(new Font("Arial", Font.BOLD, 14));
        panel.add(passwordLabel);
        passwordField.setBounds(160, 300, 200, 30);
        panel.add(passwordField);

        JLabel favoriteWordLabel = new JLabel("<html>Favorite</html> Word: <html>");
        favoriteWordLabel.setBounds(40, 350, 150, 30);
        favoriteWordLabel.setFont(new Font("Arial", Font.BOLD, 14));
        panel.add(favoriteWordLabel);
        favoriteWordField.setBounds(160, 350, 200, 30);
        panel.add(favoriteWordField);

        deleteBtn.setBounds(110, 400, 150, 35);
        deleteBtn.setFont(new Font("Arial", Font.BOLD, 14));
        deleteBtn.addActionListener(this);
        panel.add(deleteBtn);

        generalDbButton.setBounds(180, 450, 200, 50);
        generalDbButton.setFont(new Font("Arial", Font.BOLD, 14));
        generalDbButton.addActionListener(this);
        panel.add(generalDbButton);
    }

    JPanel getPanel() {
        return panel;
    }

    public void setUsernameField(String text) {
        usernameField.setText(text);
    }

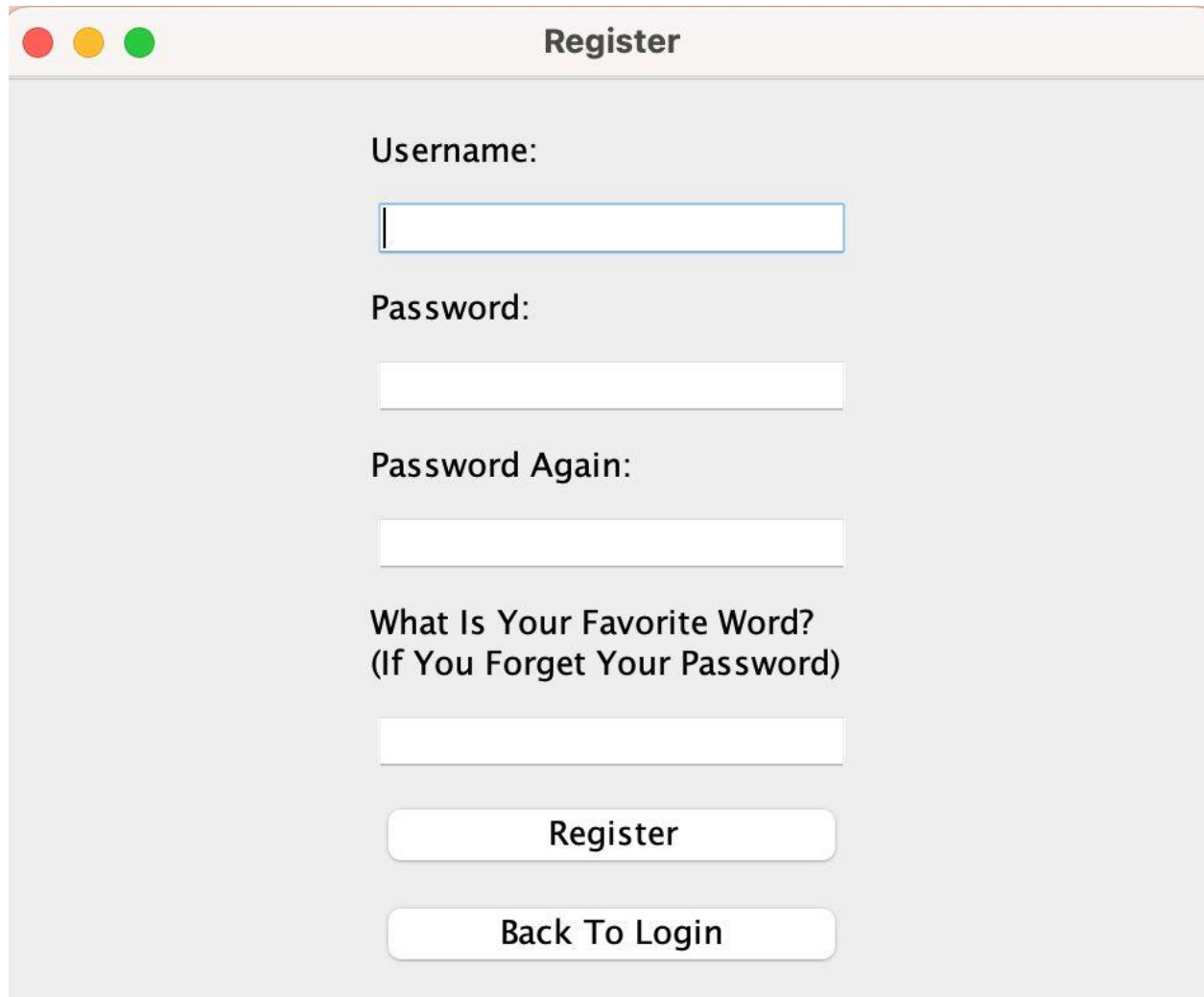
    public void setPasswordField(String text) {
        passwordField.setText(text);
    }

    public void setFavoriteWordField(String text) {
        favoriteWordField.setText(text);
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == deleteBtn) {
            this.dispose();
            String selectedUsername = usernameField.getText();
            if (!selectedUsername.isEmpty()) {
                UserControlTable userControlTable = new UserControlTable(username);
                userControlTable.deleteUser(selectedUsername);
            }
        } else if (e.getSource() == generalDbButton) {
            this.dispose();
            MyGeneralTable generalTable = new MyGeneralTable(username);
            panelForGeneralDB.disabled();
        }
    }
}
```

UI Snippets:

Register Page:

A screenshot of a web browser window titled "Register". The window has a light gray background and a title bar with three colored buttons (red, yellow, green) on the left. The form contains four input fields: "Username:", "Password:", "Password Again:", and "What Is Your Favorite Word? (If You Forget Your Password)". Below the input fields are two buttons: "Register" and "Back To Login".

Register

Username:

Password:

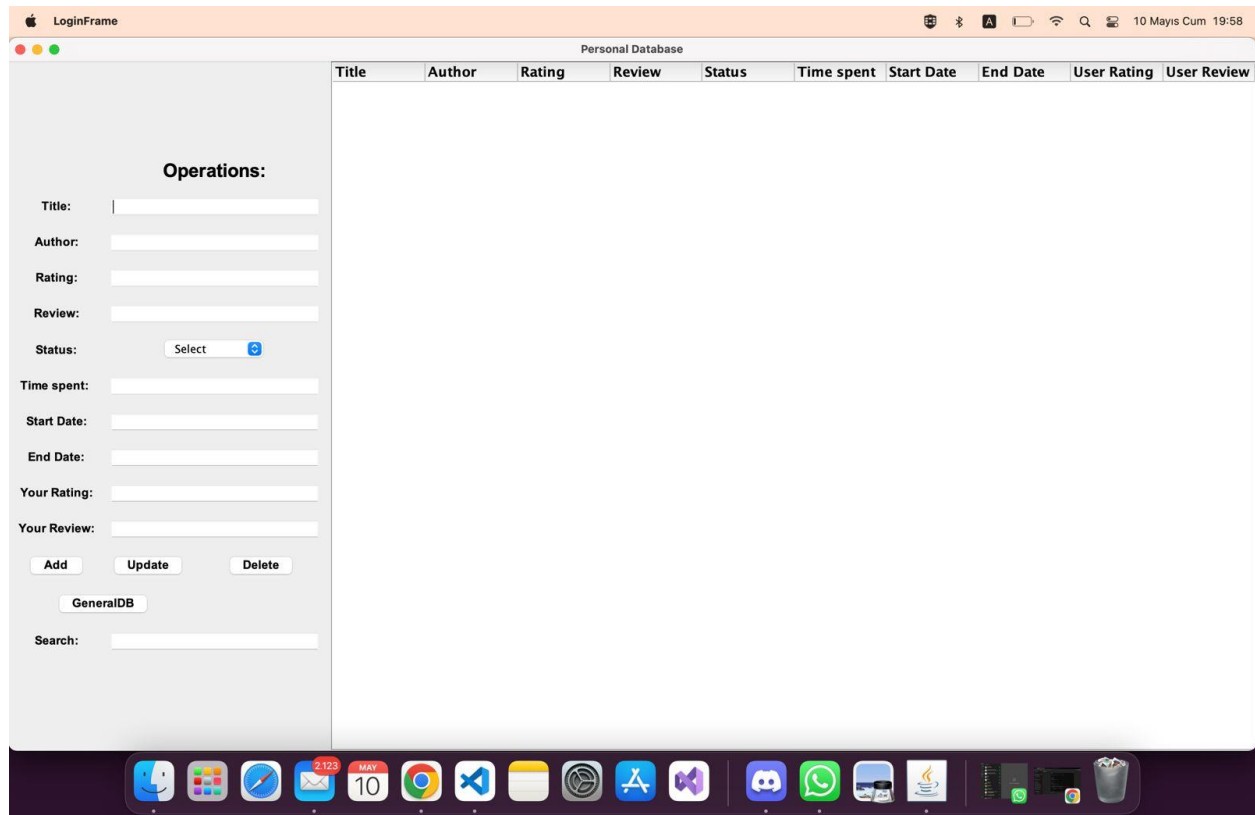
Password Again:

What Is Your Favorite Word?
(If You Forget Your Password)

Register

Back To Login

Personal Database Page:



General Database for User Page:

		Title	Author	Rating	Review
Search	<input type="text"/>	AASSADSADSADSA	Unknown	No rating	No review
		Plays	Aeschyles	No rating	No review
Operations	<input type="text"/>	Kele	Author	No rating	No review
		Pesi	Author	No rating	No review
Title:	<input type="text"/>	Playsssssa	Aeschyles	No rating	No review
		HistoRaul	Herodotus	No rating	No review
Author:	<input type="text"/>	HistoAraz	Herodotus	No rating	No review
		The Phorician Women	Euripides	No rating	No review
Rating:	<input type="text"/>	ElectraNtre	Euripides	No rating	No review
		The Peloponesian War	Thucydides	No rating	No review
Review:	<input type="text"/>	Poetics	Aristotle	No rating	No review
		Dialog	Plato	No rating	No review
Add to Personal Library	<input type="text"/>	Physics	Aristotle	No rating	No review
		Ethics	Aristotle	No rating	No review
PersonalDB	<input type="text"/>	Greek Anthology	Unknown	5.0 (1)	me
		De Animal	Aristotle	4.0 (1)	me
		On The Nature of Things	Lucretius	No rating	No review
		Parallel Lives	Plutarch	No rating	No review
		Aeneid	Virgil	No rating	No review
		Bucolics	Virgil	3.0 (1)	me
		Georgics	Virgil	No rating	No review
		Annals	Tacitus	No rating	No review
		Metamorphoses	Ovid	No rating	No review
		Heroides	Ovid	No rating	No review
		Amores	Ovid	No rating	No review
		The Twelve Caesars	Suetonius	No rating	No review
		Meditations	Marcus Aurelius	No rating	No review
		Poems	Catullus	No rating	No review
		Poems	Horace	No rating	No review
		Discourses	Epictetus	No rating	No review
		Plays	Aristophanes	No rating	No review
		Historical Miscellany	Aelianus	No rating	No review
		On the Nature of Animals	Aelianus	No rating	No review
		Argonautica	Apollonius Rhodius	No rating	No review
		Fourteen Byzantine Rulers	Michael Psellus	No rating	No review
		The Rise and Fall of the Roman Empire	Edward Gibbon	No rating	No review
		The Enneads	Plotinus	No rating	No review
		Ecclesiastical History	Eusebius	No rating	No review
		Consolations of Philosophy	Boethius	No rating	No review
		Letters	Pliny the Younger	No rating	No review
		Byzantine verse romances	Unknown	No rating	No review
		Fragments	Heraclitus	No rating	No review
		Confessions	St. Augustine	No rating	No review
		Summa Theologica	Thomas Aquinas	No rating	No review
		The Little Flowers	St. Francis of Assisi	No rating	No review
		The Prince	Niccolò Machiavelli	No rating	No review
		Divide Comedy	Dante Alighieri	No rating	No review
		Novelle	Franco Sacchetti	No rating	No review
		Icelandic sagas	Unknown	No rating	No review

General Database for Admin Page:

		Title	Author	Rating	Review
Search	<input type="text"/>	AASSADSADSADSA	Unknown	No rating	No review
		Plays	Aeschyles	No rating	No review
		Kele	Author	No rating	No review
		Pesi	Author	No rating	No review
		Playsssssa	Aeschyles	No rating	No review
		HistoRaul	Herodotus	No rating	No review
		HistoAraz	Herodotus	No rating	No review
		The Phornician Women	Euripides	No rating	No review
		ElectraNtre	Euripides	No rating	No review
		The Peloponesian War	Thucydides	No rating	No review
		Poetics	Aristotle	No rating	No review
		Dialog	Plato	No rating	No review
		Physics	Aristotle	No rating	No review
		Ethics	Aristotle	No rating	No review
		Greek Anthology	Unknown	5.0 (1)	me
		De Animal	Aristotle	4.0 (1)	me
		On The Nature of Things	Lucretius	No rating	No review
		Parallel Lives	Plutarch	No rating	No review
		Aeneid	Virgil	No rating	No review
		Bucolics	Virgil	3.0 (1)	me
		Georgics	Virgil	No rating	No review
		Annals	Tacitus	No rating	No review
		Metamorphoses	Ovid	No rating	No review
		Heroides	Ovid	No rating	No review
		Amores	Ovid	No rating	No review
		The Twelve Caesars	Suetonius	No rating	No review
		Mediations	Marcus Aurelius	No rating	No review
		Poems	Catullus	No rating	No review
		Poems	Horace	No rating	No review
		Discourses	Epictetus	No rating	No review
		Plays	Aristophanes	No rating	No review
		Historical Miscellany	Aelianus	No rating	No review
		On the Nature of Animals	Aelianus	No rating	No review
		Argonautica	Apollonius Rhodius	No rating	No review
		Fourteen Byzantine Rulers	Michael Psellus	No rating	No review
		The Rise and Fall of the Roman Empire	Edward Gibbon	No rating	No review
		The Enneads	Plotinus	No rating	No review
		Ecclesiastical History	Eusebius	No rating	No review
		Consolations of Philosophy	Boethius	No rating	No review
		Letters	Pliny the Younger	No rating	No review
		Byzantine verse romances	Unknown	No rating	No review
		Fragments	Heraclitus	No rating	No review
		Confessions	St. Augustine	No rating	No review
		Summa Theologica	Thomas Aquinas	No rating	No review
		The Little Flowers	St. Francis of Assisi	No rating	No review
		The Prince	Niccolò Machiavelli	No rating	No review
		Divide Comedy	Dante Alighieri	No rating	No review
		Novelle	Franco Sacchetti	No rating	No review
		Icelandic sagas	Unknown	No rating	No review

User Manager Page:

User Manager

Username:

Password:

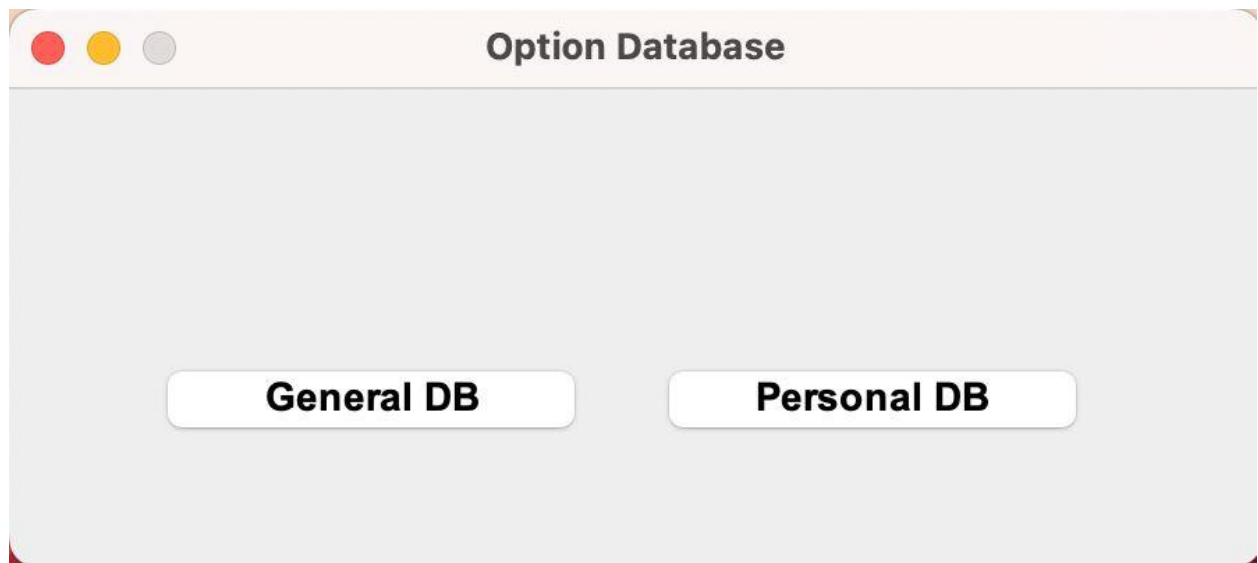
Favorite Word:

Delete User

GeneralDB

Username	Password	Favorite Word
murad	Murad2005!	murad

Option Database Page:



Login Page:



Book Library Login

Welcome Our Library

Username:

Password:

Login

Don't you have an account?

Register

Did you forgot your password?

Renew Password