import java.awt.\*;

import java.io.\*;

import java.util.HashMap;

import java.util.Map;

import javax.swing.\*;

public class QuickLinkShortenerGUI extends JFrame {

    private JTextField longUrlField, shortUrlField;

    private JTextArea resultArea;

    private Map<String, String> urlMapping;

    private static final String FILE\_NAME = "urlMappings.txt";

    public QuickLinkShortenerGUI() {

        urlMapping = new HashMap<>();

        loadUrlMappingsFromFile();

        setTitle("QuickLink Shortener");

        setSize(400, 300);

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        setLayout(new BorderLayout());

        // Input panel

        JPanel inputPanel = new JPanel();

        inputPanel.setLayout(new GridLayout(3, 2));

        inputPanel.add(new JLabel("Long URL:"));

        longUrlField = new JTextField();

        inputPanel.add(longUrlField);

        inputPanel.add(new JLabel("Short URL (optional):"));

        shortUrlField = new JTextField();

        inputPanel.add(shortUrlField);

        JButton shortenButton = new JButton("Shorten URL");

        shortenButton.addActionListener(e -> shortenUrl());

        inputPanel.add(shortenButton);

        JButton retrieveButton = new JButton("Retrieve Original URL");

        retrieveButton.addActionListener(e -> retrieveOriginalUrl());

        inputPanel.add(retrieveButton);

        add(inputPanel, BorderLayout.NORTH);

        // Result display area

        resultArea = new JTextArea();

        resultArea.setEditable(false);

        JScrollPane scrollPane = new JScrollPane(resultArea);

        add(scrollPane, BorderLayout.CENTER);

        setVisible(true);

    }

    private void shortenUrl() {

        String longUrl = longUrlField.getText().trim();

        String shortUrl = shortUrlField.getText().trim();

        if (longUrl.isEmpty()) {

            JOptionPane.showMessageDialog(this, "Please enter a valid long URL.");

            return;

        }

        if (shortUrl.isEmpty()) {

            // Automatically generate a short URL

            shortUrl = generateShortUrl(longUrl);

        }

        if (urlMapping.containsKey(shortUrl)) {

            JOptionPane.showMessageDialog(this, "This short URL already exists. Try another.");

            return;

        }

        urlMapping.put(shortUrl, longUrl);

        resultArea.append("Shortened URL: " + shortUrl + " -> " + longUrl + "\n");

        saveUrlMappingsToFile();

        longUrlField.setText("");

        shortUrlField.setText("");

    }

    private void retrieveOriginalUrl() {

        String shortUrl = shortUrlField.getText().trim();

        if (shortUrl.isEmpty()) {

            JOptionPane.showMessageDialog(this, "Please enter a short URL to retrieve the original.");

            return;

        }

        if (urlMapping.containsKey(shortUrl)) {

            String longUrl = urlMapping.get(shortUrl);

            resultArea.append("Original URL for " + shortUrl + ": " + longUrl + "\n");

        } else {

            JOptionPane.showMessageDialog(this, "No mapping found for this short URL.");

        }

    }

    private String generateShortUrl(String longUrl) {

        // Create a simple hash-based short URL

        return "short.ly/" + Math.abs(longUrl.hashCode());

    }

    private void loadUrlMappingsFromFile() {

        File file = new File(FILE\_NAME);

        if (file.exists()) {

            try (BufferedReader reader = new BufferedReader(new FileReader(file))) {

                String line;

                while ((line = reader.readLine()) != null) {

                    String[] parts = line.split("\\|");

                    if (parts.length == 2) {

                        urlMapping.put(parts[0], parts[1]);

                    }

                }

            } catch (IOException e) {

                e.printStackTrace();

            }

        }

    }

    private void saveUrlMappingsToFile() {

        try (BufferedWriter writer = new BufferedWriter(new FileWriter(FILE\_NAME))) {

            for (Map.Entry<String, String> entry : urlMapping.entrySet()) {

                writer.write(entry.getKey() + "|" + entry.getValue() + "\n");

            }

        } catch (IOException e) {

            e.printStackTrace();

        }

    }

    public static void main(String[] args) {

        SwingUtilities.invokeLater(QuickLinkShortenerGUI::new);

    }

}

OUTPUT:

