جامعة ٦ أكتوبر
October 6 University

كلية نظم المعلومات و علوم الحاسب
Faculty of Information Systems & Computer Science

Mohab Nasser Abdelkader	23013228
Menna Khaled Abdelrheem	23012753
Rahaf Alsayed Ali	23012630

Drawing & Painting Application Documentation

Project Overview

This application is a simple drawing and painting program built using Java Swing. It consists of a login system and a painting interface where users can draw basic shapes and modify the display settings.

Application Structure

The application is divided into three main classes:

1. ProjectVP - Handles the login interface and user authentication

2. PaintingFrame - Creates the main painting interface with controls

3. PaintingPanel - Implements the drawing functionality

Class Details

ProjectVP Class

This is the entry point of the application that handles the login functionality.

Key Components:

- **JFrame**: Main application window

- **JTextField**: Input fields for student name and ID

- **HashMap**: Stores valid student credentials

- **Login Logic**: Validates user credentials against stored data

Authentication Process:

1. User enters name and ID in the provided fields

2. System checks if the name exists in the database

3. System verifies if the ID matches the stored value for that name

4. If credentials are valid, the login frame closes and opens the PaintingFrame

Pre-registered Users:

- Username: "menna", ID: "127"

- Username: "mohab", ID: "128"

- Username: "rahaf", ID: "129"

PaintingFrame Class

This class creates the main interface for the drawing application after successful login.

Key Components:

- **JFrame**: Main window for painting

- **PaintingPanel**: Custom JPanel for drawing shapes

- **JButtons**: Controls for selecting shapes and clearing the panel

- **JMenuBar**: Menu system for additional functionality

Menu Options:

1. **Change Background Color**:
   - Pink
   - Cyan
   - Orange

2. **Change Font**:
   - Plain
   - Bold
   - Italic

3. **About**:
   - Shows information about the developers

Draw Controls:

- Rectangle button: Draws a filled rectangle

- Oval button: Draws a filled oval

- Line button: Draws a straight line

- Clear button: Clears the drawing panel

PaintingPanel Class

This custom JPanel implements the actual drawing functionality.

Key Components:

- **shapeCode**: Integer indicating which shape to draw

- **JLabel**: Displays welcome text with configurable font

- **paintComponent**: Override method that handles the actual drawing

Drawing Operations:

- **Oval (shapeCode 1)**: Draws a filled black oval

- **Rectangle (shapeCode 2)**: Draws a filled black rectangle

- **Line (shapeCode 3)**: Draws a straight black line

- **Clear (shapeCode 4)**: Clears the panel

Customization Methods:

- **changeLabelFont()**: Changes the font style of the welcome label

- **changeBackgroundColor()**: Changes the background color of the panel

Program Flow

1. **Startup**:
   - Application launches showing the login screen
   - User must enter valid credentials to continue

2. **Authentication**:
   - System validates entered credentials
   - If valid, launches the painting interface
   - If invalid, shows appropriate error message

3. **Painting Interface**:
   - User can select shapes to draw from buttons
   - Menu options allow changing background color and text font
   - About menu provides information about the developers

4. **Drawing Process**:

    o   Selecting a shape button sets the shape code

    o   The paintComponent method renders the selected shape

    o   Clear button resets the drawing panel

Technical Implementation Details

Event Handling

The application uses ActionListener interfaces to handle:

- Button clicks

- Menu item selections

- Login form submission

Graphics Rendering

- Java's Graphics class is used for shape rendering

- The paintComponent method is overridden to implement custom drawing

UI Layout Management

- The login screen uses null layout for precise component positioning

- The painting frame uses BorderLayout and FlowLayout for organized UI

Data Storage

- Student credentials are stored in a HashMap for quick lookup

- Key: Student name (lowercase)

- Value: Student ID

Error Handling

The login system includes validation for:

1. Empty name field

2. Empty ID field

3. Unregistered student names

4. Incorrect ID for registered names

Each error displays an appropriate message dialog to guide the user.