Project Overview

This application is a standalone multimedia tool built using Java and JavaFX, designed for local image browsing, annotation, editing, and creative content generation. Users can annotate photos, perform image edits, and generate videos or artistic mosaics from selected pictures. The project was developed as a collaborative group assignment for the WIG3003 module.

Key Objectives

- Allow users to browse and view all local image files from a designated folder
- Provide annotation capability for each photo using text notes
- Offer a wide range of image editing features including brightness adjustment, contrast enhancement, grayscale conversion, rotation, cropping, and undo/redo operations
- Enable users to select multiple images and:

  Generate a video from the selected photos, with optional text and graphical overlays

  Create a mosaic using the selected images, arranged in either a heart or a star shape
- Ensure all data is handled and saved locally on the user's machine without requiring an internet connection

Task distribution

Member	Assigned Tasks	Technical Scope
Harry（A）	Main UI design + Photo browsing + Annotation feature (including ♥ icon)	Swing UI design, image display, annotation handling, icon overlay
Nowesh（B）	Image editing features (brightness, contrast, grayscale, rotate, etc.) + Image data management	Image processing logic, file/database handling, caching
Leen（C）	Video creation (photo stitching, adding text/graphics) + Video playback control	Java media APIs, timers, image-to-video conversion
Meilin（D）	Mosaic generation + Project integration + Demo video editing	Image composition logic, integration/debugging, basic video editing tools

Workflow

Date Range	Task	Responsible Member	Notes
May 1 – May 3	Team discussion, UI sketching, task distribution	All Members	Define module boundaries and communication interfaces
May 4 – May 9 (parallel)	UI layout & image browsing development	Member A	Provides base for other features
	Image editing functions (brightness, contrast, grayscale, etc.)	Member B	Independent logic
	Initial video creation module (photo stitching, text/graphics)	Member C	Can use dummy images for testing
	Mosaic generation logic	Member D	No dependency, can proceed alone
May 9 – 12 (parallel)	Annotation system + ♥ indicator	Member A	Integrate with image viewer

	Finalize image editing + image storage system	Member B	Add file/database saving
	Complete video features + playback control	Member C	Link playback with generation
	Mosaic feature testing + UI integration	Member D	Connect to UI and ensure display
May 13 – 15 (collaborative)	Feature integration, UI refinement, full testing	All Members	Sync all modules, resolve interface issues
May 16 – 17	Debugging, UI polishing, final tweaks	All Members	Based on integration test results
May 18	Record demo video + video editing	Member D (lead), others assist	Use latest version for screen recording
May 19	Write ReadMe file, assign member roles, write final report	All Members	Each member writes their part
May 20	Final check, zip & submit	All Members	Ensure all files are included

Software Requirements

To compile and run this application successfully, the following tools and libraries must be installed:

- Java Development Kit (JDK) version 17 or later
- JavaFX SDK (tested with JavaFX 18)
- An Integrated Development Environment (IDE) such as IntelliJ IDEA or Eclipse
- JavaCV library for video generation (includes FFmpeg and OpenCV bindings)
- A working internet connection is not required during runtime

> Ensure JavaFX libraries are properly linked in your module path, and native libraries for JavaCV are configured appropriately.

How to Set Up and Run the Application

1. Clone or download this repository to your local machine.
2. Open the project in your preferred IDE (such as IntelliJ IDEA).
3. Ensure the directory named `photos/` exists in the root of your project. Place image files (`.jpg`, `.jpeg`, `.png`) into this folder.
4. Run the `Main.java` file to launch the application.
5. Follow the on-screen instructions to navigate through features.
6.

Feature Descriptions

Main Interface

- Automatically displays thumbnails of all images located in the `photos/` directory.
- Each image is shown in a grid layout. Annotated images are marked with a red heart (❤️) symbol.

- Clicking on an image opens the photo in a detailed view page.

Photo Annotation

- Users can annotate each image by clicking the “Edit Your Note” button.
- An annotation dialog allows users to enter and save text notes.
- Notes are saved locally in a `.txt` file with the same name as the image.
- Once a note is saved, it is displayed below the corresponding image in the detail view.

Image Editing

By clicking the “Edit Your Photo” button, users can open an editing interface where the following operations are supported:

- Increase brightness
- Enhance contrast
- Apply grayscale filter
- Crop the image (fixed or interactive)
- Rotate the image by 90 degrees
- Undo or redo changes
- Save the edited image

Image Selection and Batch Processing

- The Select Page displays all available images with checkboxes.
- Users can individually select images or click “Select All” to toggle all selections.
- Once at least one image is selected, two options become available:
    - “Create Your Video” for video generation

- “Create Image Mosaic” for mosaic generation

Video Creation Module

- Automatically generates a short video using the selected images.
- Users can:
    - Add a text overlay that appears in each video frame
    - Select a graphical image (logo or icon) to overlay on the video
- Images are resized and encoded using the FFmpeg encoder via JavaCV
- Output format: MP4
- A video preview is shown using JavaFX MediaPlayer after generation

🧩 Mosaic Generator

- Generates a single artistic image mosaic using the selected photos
- The system randomly selects either a heart shape or a star shape as the template
- All selected images are resized into small tiles and arranged to fill the shape
- The mosaic is generated using `Graphics2D` and rendered to a `BufferedImage`
- The result is displayed in the interface using an `ImageView`

Project Structure

```
project-root/
├── photos/                        # Folder containing user images
└── src/org/example/
    ├── Main.java                  # Entry point of the application
    ├── MainPage.java              # Home screen and gallery viewer
    ├── SelectPage.java            # Photo selection interface for batch processing
    ├── PhotoPage.java             # Photo detail viewer with note editor
    ├── EditDialog.java            # Note editing pop-up dialog
    ├── EditPhotoPage.java         # Full image editor UI
    ├── VideoCreatorModule.java    # Video generation logic using JavaCV
    └── MosaicPage.java            # Mosaic generation using Graphics2D
```

Notes and Limitations

- Ensure that valid image files are placed in the `photos/` directory before launching the application.
- Video creation requires JavaCV and FFmpeg to be correctly configured.
- Mosaic shapes (heart and star) are randomly selected at runtime.
- All operations are local and no data is transmitted over the internet.