

Figurio

"Boost your projects with figures"

Web Application for Image Editing for Technical Documents



Demo Video

Author: Pavol Humený

Contact: pavol.humeny@gmail.com

Academic Year: 2025/2026

Supervisor : prof. Ing. Adam Herout, Ph.D.

Motivation

- Poor-quality images in technical documents
- Time-consuming screenshot and diagram editing
- Complex graphic editors
- Missing specialized tools for image editing

Figurio

- Web application available in the browser
- No installation required
- Simple and intuitive interface
- Designed for fast screenshot and diagram editing

Key Features

- Local image processing ensures privacy
- Support for vector PDF editing
- Specialized tools for academic image editing
- Real size image preview

Image Import

- Supported input formats: PNG, JPG, WebP, and PDF
- The file format is detected after loading
- Raster images are loaded into a canvas
- PDF documents are processed using pdf.js

Operation Processing

- Tools store operation definition and parameters
- Operations are stored in a sequential list
- The central pipeline loads the current image data
- Based on the operation type, the appropriate processing module is selected, and the result is stored in the store

Rendering

- Module watch changes in the application state
- When the state changes, the current image is rendered
- Rendering includes raster data, vector layers and presentation frames

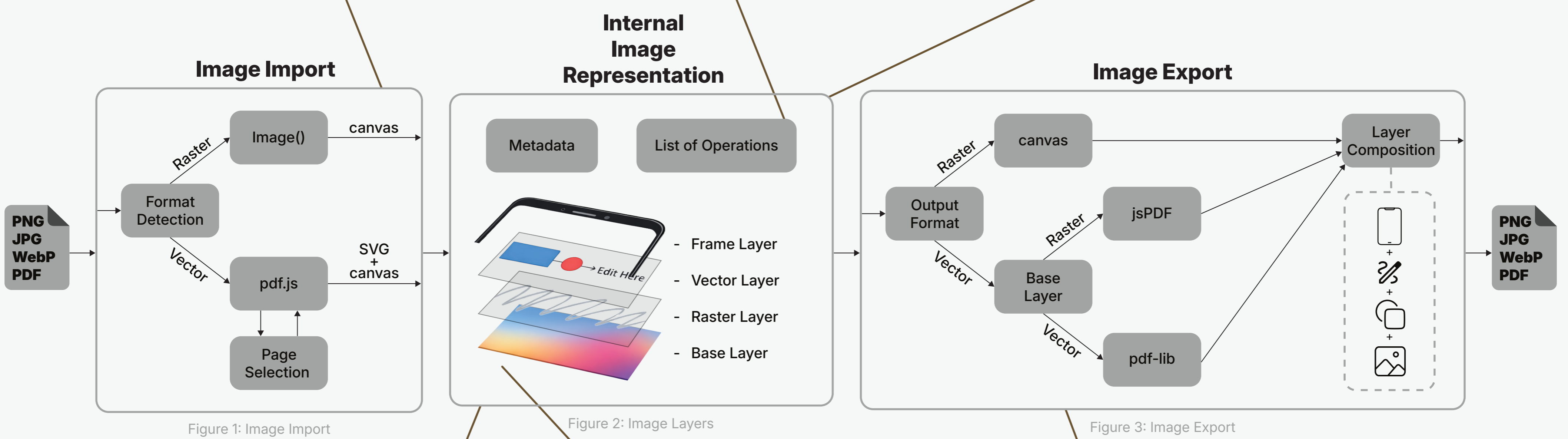


Figure 1: Image Import

Figure 2: Image Layers

Figure 3: Image Export

Internal Image Representation

- The central application state is implemented as a Pinia store
- It stores the current image, metadata, and the list of operations
- It provides data for processing and rendering

Layered image representation

- The final image is composed of multiple layers
- This model allows independent editing of each layer
- Frame Layer – presentation frame, device mockup
- Vector Layer – vector objects, annotations and text
- Raster Layer – brush drawing and raster edits
- Base Layer – original image or document content

Image Export

- The user selects the output format and export settings
- Layers are merged into the final image
- Raster formats use canvas export
- PDF is generated using jsPDF and pdf-lib

Crop

- Crop the image and remove unwanted parts
- Manual free cropping or cropping with the original image aspect ratio
- **Fit crop** – detect image content automatically

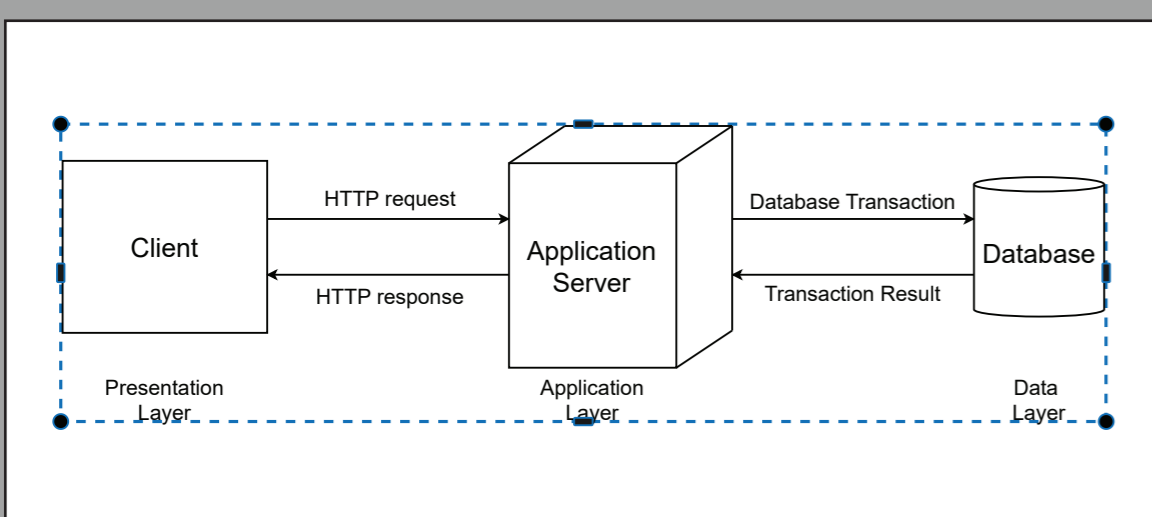


Figure 4: Crop Tool

Magnify Area + Blur Area

- Zoom in on a selected part of the image
- Blur a selected part of the image
- Highlight key elements and hide sensitive data in the image

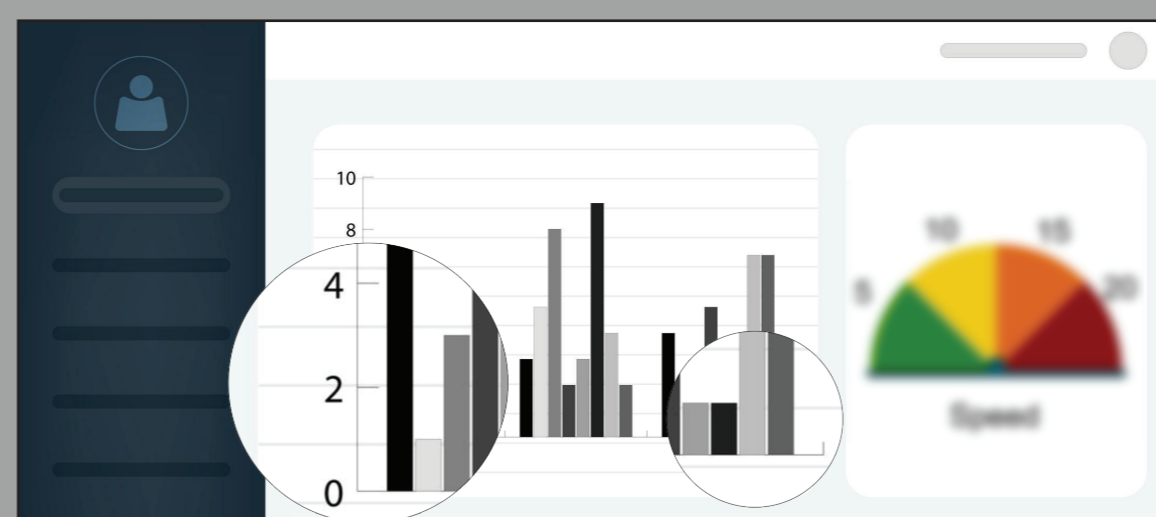


Figure 5: Magnify Area and Blur Area Tool

Frames

- Place the image into a presentation frame
- Available frames: simple frame app window, mobile mockup, and code editor
- Suitable for presenting user interfaces

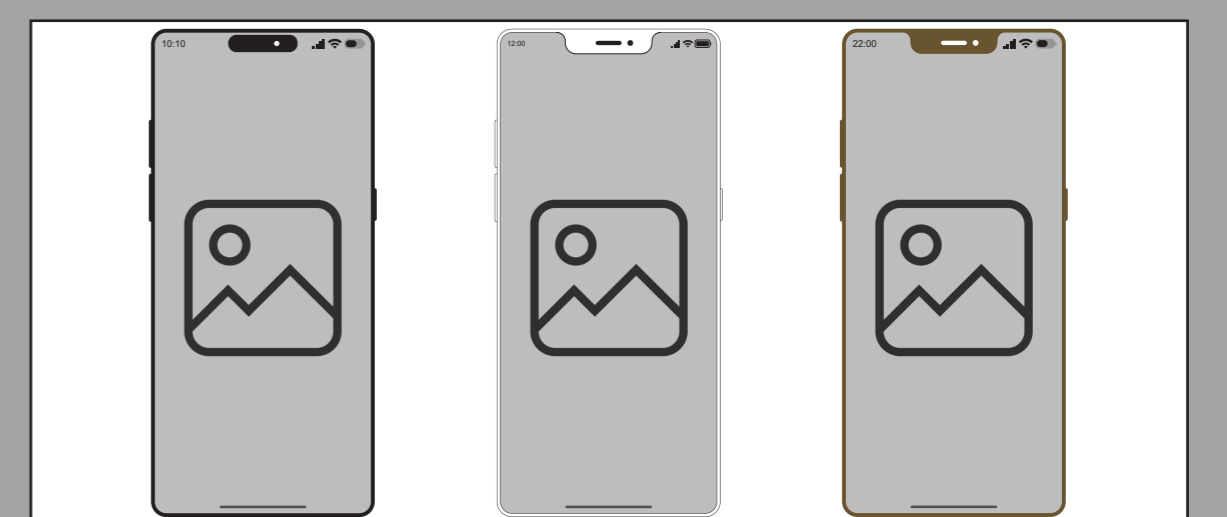


Figure 6: Frame Tool

Available Tools

- | | | | | |
|-----------------|--------|--------------------|--------|-----------|
| Noise Detection | Select | Crop | Rotate | Grayscale |
| Magnify Area | Text | Presets | Flip | Frames |
| Blur Area | Shape | Brush | Resize | |
| | | Background Removal | | |



<https://pavol-humeny.github.io/figurio>