📘 PDFFlow Documentation

Overview

PDFFlow is a fast, privacy-first, browser-based PDF toolkit that allows users to perform common PDF operations without uploading files to any server. All processing happens locally in the user's browser, ensuring complete data privacy and security.

The project is designed to be lightweight, dependency-minimal, and easy to use on both desktop and mobile devices.

🎯 Goals & Philosophy

- 🔒 **Privacy by design** – No uploads, no servers, no tracking
- ⚡ **Performance** – Instant processing using client-side JavaScript
- 🌐 **Accessibility** – Works directly in the browser, no installation
- 🧩 **Simplicity** – Minimal UI, clear actions, predictable behavior

---

🧱 Architecture

PDFFlow follows a **pure client-side architecture**:

- No backend server
- No databases
- No authentication
- No external APIs for file handling

All PDF operations are executed using JavaScript libraries directly in the browser.

🛠️ Technology Stack

Technology	Purpose
HTML5	Page structure
CSS3	Styling & responsive UI
Vanilla JavaScript	Application logic

Technology	Purpose
PDF-lib	PDF manipulation (split, merge, numbers)
jsPDF	PDF generation from images

📁 Project Structure

```
PdfFlow/
├─ index.html          # Main entry point
├─ css/
│  ├─ style.css        # Core styling
│  └── modal.css       # Modal & UI components
├─ js/
│  ├─ main.js          # App initialization
│  ├─ pdfOperations.js # PDF processing logic
│  └── uiHandlers.js   # UI event handling
── favicon.ico         # Website favicon
└── README.md          # Project documentation
```

🔧 Core Features

📄 Split PDF

- Extract all pages as individual PDFs
- Extract specific page ranges (e.g. 1-3, 5, 7-9)
- Automatically names output files

📎 Merge PDF

- Combine multiple PDFs into one
- Drag-and-drop file ordering
- Preserves original quality

🗜️ Compress PDF

- Reduces file size while keeping visual quality
- Displays compression results
- One-click operation

🔢 Add Page Numbers

- Multiple position options (top/bottom, left/center/right)
- Sequential numbering
- Non-destructive to original content

🖼️ Image to PDF

- Convert JPG and PNG images to PDF
- Supports multiple images at once
- Maintains original resolution and layout

🔐 Privacy & Security

PDFFlow is built with **privacy as a core principle**:

- ✅ Files never leave the user's device
- ✅ No uploads to any server
- ✅ No analytics, cookies, or trackers
- ✅ No account or login required
- ✅ Works offline after first load

This makes PDFFlow suitable for handling **sensitive or confidential documents**.

🌐 Browser Compatibility

Supported browsers:

- Chrome / Edge (v90+)
- Firefox (v88+)
- Safari (v14+)

- Opera (v76+)

Mobile support:

- iOS Safari
- Android Chrome
- Mobile Firefox

🚀 Deployment

PDFFlow is deployed using **GitHub Pages**.

Live URL:

https://poovizhivm.github.io/PdfFlow/

Deployment requires:

- Static files only
- index.html located in the repository root
- Relative asset paths (no absolute /css/... paths)

🧪 Local Development

To run PDFFlow locally:

git clone https://github.com/Poovizhivm/PdfFlow.git

cd PdfFlow

Then either:

- Open index.html directly in a browser
  OR

python -m http.server 8000

Visit:

http://localhost:8000

🤝 Contributing

Contributions are welcome!

You can help by:

- Reporting bugs
- Suggesting new features
- Improving UI/UX
- Refactoring code
- Improving documentation

Contribution steps:

1. Fork the repository
2. Create a new branch
3. Make your changes
4. Submit a pull request

📄 License

This project is licensed under the **MIT License**.

You are free to:

- Use
- Modify
- Distribute
- Use commercially

📬 Contact

- **Author:** Poovizhivm
- **Email:** poovizhivm@gmail.com
- **GitHub:** https://github.com/Poovizhivm
