

PDF Merger

Documentation

Source code available under licenses can be found at:

<https://codecanyon.net/item/pdf-merger-merge-rotate-organize-delete-preview-pdf-all-in-one-react-nextjs-web-application/42741922>

Table of content

Introduction	3
Features	3
Getting Started	4
System requirements.....	4
Steps to run the project locally.....	4
Accessing the web app	4
Selecting PDF files	5
User Guide	5
Customization	6
Q&As	7
Troubleshooting.....	11
Technical Documentation	12
Technology Stack	12
Project Structure.....	12
Deployment	14
Hosting Requirements	14
Recommended Hosting services.....	14
– Vercel	14
– VPS (Virtual Private Server):	14
– Digitalocean Droplet	14
– Digitalocean App platform	15
– Render.com.....	15
Google analytics.....	15
Multiple Languages Support.....	15
Limitations	16
Source code	16
Support	17

Introduction

PDF merger is a fully featured client-side web application that enables you to not only merge multiple PDFs into one file but also change the order of the pages, delete pages, rotate pages and see a preview of the merged file. It only takes a few seconds to merge PDF files together, regardless of file size. The process of merging PDF files is done on the client side (directly in the browser). No backend is used in this application and PDFs are not stored in server nor in a database.

Features

- Fast PDF files merging.
- Rotate PDF Pages.
- Delete PDF Pages.
- Organize PDF pages.
- PDF pages viewer; display preview of the merged PDF file pages.

Additionally:

- The maximum file size the tool can accept as input can be changed.
- Right to left support.
- Security guaranteed (All files are processed directly on your browser).
- Compatible with All Browsers.
- Responsive.
- SEO Friendly.
- Google Analytics Ready.
- Multiple languages support.
- About, Terms and Condition, Privacy and Contact Pages.

Getting Started

System requirements

To run PDF Merger web app, you will need a device that meets the following minimum system requirements:

- **Operating System:** Windows, macOS, or Linux.
- **Server Requirements:** The server must support [Node.js](#) and have a CPU with 2 to 4 cores and RAM of 2 to 8 GB.
- **Client Requirements:** The app can be accessed through any modern web browser with JavaScript enabled.
- **Database:** Not required as no database is used in the app.

Steps to run the project locally

To run this project locally you need to have installed:

- Your favorite code editor (Example **Visual code**. [Download link](#))
- NodeJS ([Download link](#))

After installing NodeJS:

1. Unzip the project file.
2. Open your code editor, example visual code, click on file => open folder and select project folder.
3. Click on Terminal => new Terminal and then type: npm install (this will install all the necessary dependencies for the project)
4. To run the project type: npm run dev
5. Open your browser and visit this URL: <http://localhost:3000>

Accessing the web app

To access PDF Merger web app, simply navigate to the website URL (If you are running the app on your local machine, navigate to: <http://localhost:3000>) using your preferred web browser. You can access PDF Merger web app on any device

with an internet connection, including desktops, laptops, tablets, and smartphones.

Selecting PDF files

To select file(s), open the tool and simply click on the "Choose File" button and select the file you want to upload. You can also drag and drop your PDF file directly onto the web app interface. PDF Merger tool allows multiple files selection. Once you have selected your file(s), you can begin using the selected tool's features to edit your files.

User Guide

Welcome to the user guide for the PDF Merger web app. This guide will walk you through the main features of the app and provide step-by-step instructions on how to use them.

Description:

This tool allows users to merge multiple PDF files into a single document. The tool allows users to select and process one or multiple files at the same time. Users can also rotate pages, delete pages, and reorganize the order of pages as needed. The tool includes a PDF viewer to preview the merged document.

How to use Merge PDF tool:

To merge multiple PDF files into a single document, select the "Merge PDF" tool and follow these steps:

1. Click on the "Choose File" button and select or drag and drop the PDF file(s) into the tool.
2. Click on the "Add File" button to add additional PDF files to the merge queue.
3. Arrange pages in the desired order by dragging and dropping them.
4. Rotate or delete pages, if needed.
5. Click 'Save and Download' to combine and download your PDF.

Customization

The PDF Merger app has been designed with customization in mind. We understand that different clients have different needs and preferences, so we have made it easy for you to customize certain aspects of the app to fit your specific requirements. Here are the options you can change:

- **title:** The title of the tool as it appears on the corresponding buttons for each tool.
- **description:** A brief description of what the tool does. The description is used to showcase each tool on the index page when displaying the list of all tools.
- **icon:** The icon that appears next to the tool's title on the corresponding buttons.
- **acceptedInputFileMaxSize:** The maximum file size (in bytes) that the tool can accept as input.
- **newFileNameSuffix:** The suffix that will be appended to the filename of the output file generated by the tool.

To make changes to these settings, you can modify the **'toolsData'** object literal found in **useToolsData.js** file located in the **hooks folder**. This file contains data for all the tools in the app, making it easy to modify settings across the board. For example, to adjust the maximum file size that a tool can handle, simply edit the **'acceptedInputFileMaxSize'** parameter for that tool in the **'toolsData'** object.

By customizing these aspects of the PDF Merger app, you can tailor it to better fit your specific use case.

Note:

It's important to note that while you can customize certain aspects of the PDF Merger app, there are some attributes that should not be changed for the app to function correctly. These include:

- **key:** This is a unique identifier for each tool and is used internally by the app to reference the tool. Changing this value could cause errors in the app.

- **acceptedInputMimeType:** This specifies the MIME type of the file that the tool can accept as input. Changing this value could cause the tool to fail when trying to process certain types of files.

It's important to exercise caution when making any changes to the 'toolsData' object to ensure that the app continues to function correctly.

Q&As

In case your question isn't answered in the FAQ section, please email me directly your question at dosana.author@proton.me

Question 01: What are all the available pages?

Answer 01: Available pages in PDF Merger App are:

- Index page (<http://localhost:3000/>).
- About page (<http://localhost:3000/about>).
- Privacy page (<http://localhost:3000/privacy-policy>).
- Terms and conditions page (<http://localhost:3000/terms-of-use>).
- Contact Page (<http://localhost:3000/contacts>).
- 404 Error Page.

Question 02: Is PDF Tool compatible with all browsers?

Answer 02: PDF Merger is compatible with any modern web browser with JavaScript enabled.

Question 03: Is PDF Tool Responsive?

Answer 03: Yes, you can access PDF Merger web app on any device with an internet connection, including desktops, laptops, tablets, and smartphones.

Question 04: Is PDF Tool SEO Friendly?

Answer 04: Yes.

Question 05: Does PDF Tool support multiple languages?

Answer 05: Yes.

Question 06: Does PDF Merger application require subscription to third-party API?

Answer 06: No, PDF Merger does not require any subscription to a third-party API.

Question 07: Does PDF Merger application use any third-party API?

Answer 07: No, PDF Merger does not use any third-party API. The process of merging PDF files is done on the client side (directly on the browser).

Question 08: How can I change the maximum size for the selected files in each tool?

Answer 08: Please read Customization section the documentation.

Question 9: Where can I add meta data in pages?

Answer 9:

- If you would like to add a meta data in head tag for all pages, you can add it in <Head> component in _app.js file located in pages folder.
- Anything you add in _app.js will automatically be added to all the pages that exist.
- If you would like to add meta data in the head tag in a specific page, you can add it in **Head** component of that page.
- Any meta data you add in any page only will be added to only that page.

Question10: Where is the Nav bar code?

Answer10: The nav component is in Nav.js file located in components folder.

Question11: Where is the footer code?

Answer11: The footer component is in Footer.js file located in components folder.

Question12: Where can I change the main title and description of the page for each page?

Answer12: Main title and description can be changed in header section in each page.

Question13: Where to change the existing CSS styles and add custom styles?

Answer13:

- You can find all CSS files in the styles folder.
- You can find global styles in **global.css** file in styles folder, here you can add styles that you will use globally. It is imported in **_app.js** file in pages folder using this line:
`import "../styles/global.css";`
- I have also used some styles from bootstrap you can find its import line in **_app.js** file in pages folder. It is imported using this line:
`import "bootstrap/dist/css/bootstrap.css";`
- To add/change styles in a specific component. You open the file of that component example **UploadAreaFormStep.js** and you look for the import line, which is importing the styles, in this case it is the line:

`import styles from "../styles/UploadContainer.module.css";`

here **UploadContainer.module.css** is the name of the CSS file from where the styles are imported. You can find it in **styles folder**, and you can edit existing styles there. Please note that if you edit or add styles in **UploadContainer.module.css** file, they will be applied only in the components where it is imported.

In **UploadAreaFormStep.js** you can use the imported styles by writing **styles.the_name_of_the_class**. Like this:

```
className={` ${styles.toolbox} py-0 mt-0 ${styles.is_upload} `}
```

py-0 mt-0 here are imported from **global.css** in **_app.js** file, since they are global styles, you can use them directly.

Question14: Where can I place my robots.txt file?

Answer14: You can place your robots.txt file in the public folder.

Question15: Where can I place my sitemap.xml file?

Answer15: You can place your sitemap.xml file in the public folder.

Question16: Where can I place my google search console HTML file?

Answer16: You can place your google search console html file in public folder, or if you used meta tag you can go to pages folder and add it in `_app.js` in **Head** component.

Question17: Where can I change the title and description tags for each page?

Answer17: Title can be changed in **Head** component of each page.

Question18: Where can I place Canonical and Alternate tags?

Answer18: You can place canonical and alternate tags in the **Head** component of each page.

Question19: Where can I change the default language?

Answer19:

- Default language can be edited in the file `next.config.js`. in **defaultLocale: "en"**, replace **en** with the code of the language that you want to put as default example **"pt"** for Portuguese.
- Also, in `LanguageCountryFlag.js` in components folder, change **"en"** with the code of the language that you want to put as default in **return flags[locale] || flags["en"]**.

Question20: Where can I change flags?

Answer20: URL of the flags images can be changed in `LanguageCountryFlag.js` in components folder.

Question21: Where can I change logo text?

Answer21:

Logo text can be found in **Nav.js** and **Footer.js** files in the **components** folder. Search for the words "**PDF**" or "**Tools**" and replace it with your Logo text.

Question20: How to add Google Analytics code?

Answer20:

To learn more about integrating Google Analytics, please refer to the "Google Analytics" section in the documentation.

Question21: Do you provide Google AdSense integration?

Answer21:

We regret to inform you that we do not offer Google AdSense code integration as part of our services.

Troubleshooting

If you encounter any issues while using our web app, please refer to the following troubleshooting tips:

- If you are having trouble uploading a PDF file, please make sure that your file is in a supported format and that it meets the size limitations.
- If you are experiencing slow performance, please check your internet connection and make sure that your device meets our system requirements.

If they're unable to resolve an issue on your own, please email me at dosana.author@proton.me

Technical Documentation

PDF Merger app is built using **Next.js**, a popular React framework, and is designed to provide users with a variety of features to manipulate PDF files.

Technology Stack:

- Programming language: [JavaScript](#)
- Framework: [Next.js](#) (Version 12.1.6)
- Software: [Node.js](#) (version 18.18.0 LTS) ([Download Link](#)).

Project Structure:

1- Components folder

This folder contains the costume components used in this app.

2- Hooks folder

This folder contains the costume hooks used in this app.

3- Helpers' folder

This folder contains JavaScript code used in this app.

4- Pages folder

This folder contains pages and Api folder containing Api endpoints.

Pages:

Every page in next.js opens with a specific router path in the web browser. You access a page with the file name. (Example: You can access about page in a browser by typing URL <http://localhost:3000/about>).

5- Public folder

- This folder contains public folders and files.
- Locales folder contains translations which are stored in. Json format you can change their content by replacing the initial text value with your text.
- Img folder contains images used in the project.

6- Styles folder

This folder contains the styles used in this app. Used [CSS Modules\(Link\)](#) and followed the [name].module.css file naming convention to add [Component-Level CSS\(Link\)](#) for each component.

- Common CSS between components and index page CSS is in **global.css**
- Some bootstrap classes are also used in this project.

7- next-i18next.config.js file

This file contains the list of all the available languages.

8- Next.config.js file

This file contains custom configuration of Next.js application as well as the environment variables.

Environment variables:

NEXT_PUBLIC_GA_ID: For Google analytics Tracking ID

For more information about Next.config.js, please visit Next.js official doc: <https://nextjs.org/docs/api-reference/next.config.js/introduction>

For more information about Environment-variables, please visit Next.js official doc: <https://nextjs.org/docs/api-reference/next.config.js/environment-variables>

9- Package.json file

This file holds important metadata about the project and its dependencies. For the full list of the libraries used in this project, please refer to **the Technology Stack** section in the technical documentation.

10- Dockerfile file

This Dockerfile is used to build a containerized environment for running PDF Merger application.

Deployment:

Hosting Requirements

Before choosing a hosting provider, it's important to check that the hosting provider supports Nodejs and Next.js framework. You can do this by checking their website, contacting their support team for more information, or reviewing their documentation.

Recommended Hosting services

Deployment steps will depend on which hosting Provider you choose to use. Since there are numerous hosting services out there and getting the project up and running can depend on many variables that are specific to the hosting environment. We have compiled a list of recommended hosting services that we have tested. We have mentioned deployment tutorial articles for each of the recommended hosting services.

- **Vercel:** You can follow this tutorial to learn how to deploy this application on Vercel (<https://vercel.com/>): <https://nextjs.org/learn-pages-router/basics/deploying-nextjs-app/deploy>
- **VPS (Virtual Private Server):**

To deploy the app on VPS using Docker follow the same steps in this video: https://youtu.be/4XaaHjR_F4

Link to commands gist: [Deploying PDF Tools Nextjs app On VPS using Docker, FileZilla, Nginx and Let's Encrypt SSL Certificates \(github.com\)](#)

PS: In the video I have used [Digitalocean](#) droplets, but the deployment steps are the same for every VPS. Once you have the necessary information such as the VPS IP address, login password, and domain name, you can follow this video tutorial to deploy your application on your VPS.

- **Digitalocean Droplet:** To deploy the app on [digitalocean](#) **Droplet** watch this video tutorial: https://youtu.be/4XaaHjR_F4

Link to commands gist: [Deploying PDF Tools Nextjs app On VPS using Docker, FileZilla, Nginx and Let's Encrypt SSL Certificates \(github.com\)](#)

- **Digitalocean App platform:** To deploy the app on [digitalocean](#) App platform watch this video tutorials:
https://youtube.com/playlist?list=PLIkR3jDFtZOSLihkL25S_t59Bw6oGhLwM
- **Render.com:** If you would like to deploy the app in Render.com (<https://render.com/>), follow steps explained in section "Deploy as a Node Server": <https://render.com/docs/deploy-nextjs-app#deploy-as-a-node-server>

If you would like to use a different hosting service, please contact us via email at dosana.author@proton.me. We will assist you in determining if it is possible to deploy the app on the selected hosting service and provide any necessary guidance.

Google analytics

The script for Google Analytics is already in place. All you need to do is add your google analytics id code. For that, open **next.config.js** file, then search for **NEXT_PUBLIC_GA_ID** and add your ID as value of **NEXT_PUBLIC_GA_ID** like this:
NEXT_PUBLIC_GA_ID: "your_google_analytics_tracking_id",

Note:

To start collecting analytics data, you simply need to deploy the application on a server with the Google Analytics code.

Multiple Languages Support

- The translations are in the **locales** folder located in the **public** folder.
- Translations are stored in .Json format you can change their content by replacing the initial text with your text.
- For more information about how internationalization is configured, please watch this video: [Full walkthrough](#). When reaching the section where it is explained how to add text strings, you can follow same steps to add new text on your pages.
- The languages modal is in **LangModal.js** file in **components** folder.
- The list of all the available languages (languages codes) can be found in **next-i18next.config.js**
- Using Nextjs Static Generation, a version of each page will be generated for each language.

- For more information you can check the explanations in the section "Dynamic Routes and getStaticProps Pages" following this [Link](#).

Limitations

In this section, we will discuss the limitations of PDF Merger application. It's important to be aware of these limitations to ensure that you can make informed decisions when using the application.

▪ Limitations in PDF pages previews:

In some rare cases, certain characters in pdf pages thumbnails might not show. This can happen when the font used in the pdf document is not supported by the library used to display the previews (React-pdf Library). Please note that this limitation will not affect the actual document after download.

▪ Limitation in opening protected files:

If PDF file has any kind of passwords set in it. PDF Merger application will consider it protected and won't be able to process it until the password is removed.

By being aware of these limitations, you can use our PDF Merger application more effectively and make informed decisions about its usage. If you have any questions or concerns about these limitations, please feel free to contact us for assistance.

Source code

Source code available under licenses can be found at:

<https://codecanyon.net/item/pdf-merger-merge-rotate-organize-delete-preview-pdf-all-in-one-react-nextjs-web-application/42741922>

Support

If you have any questions or concerns about the application, please feel free to contact us for assistance.

- Contact Information: dosana.author@proton.me