

USING NEXT-GEN IMAGE FORMAT TO ENHANCE YOUR APP'S PERFORMANCE (REACT)

Problem: Slow loading time on an application's pages.

Introduction

There are a number of possible reasons why the pages on your application may load slowly. This guide focuses on using WebP versions of your images to help your applications load faster. This guide was created alongside a React application, but the ideas in this guide can apply to other applications and websites besides React.

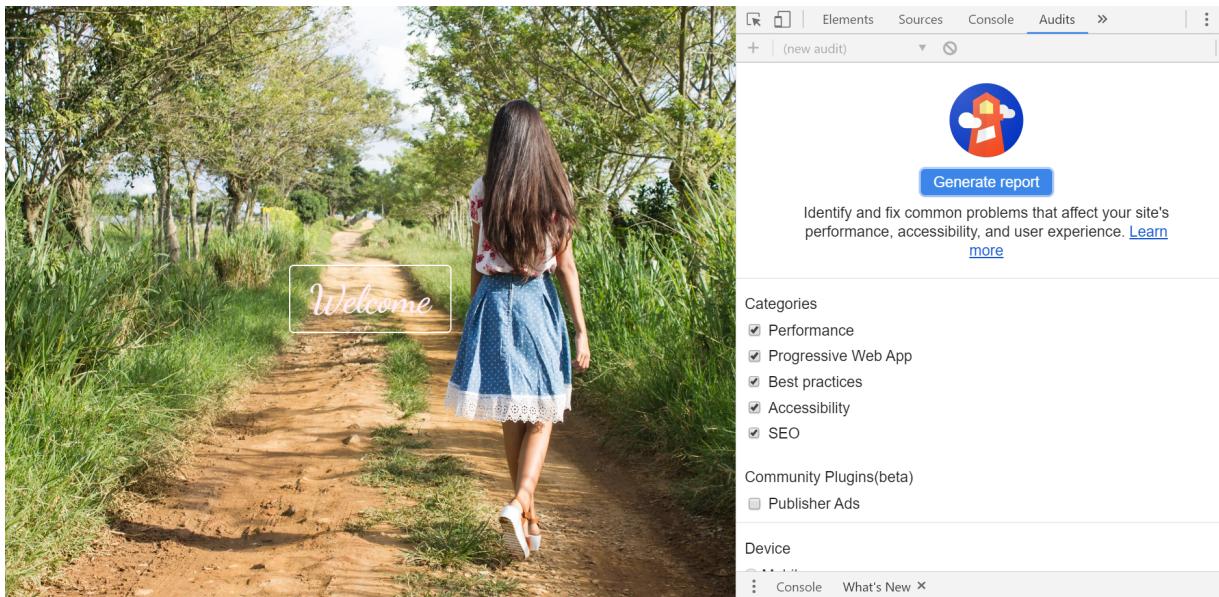
According to a web.dev article [here](#), “JPEG 2000, JPEG XR, and WebP are image formats that have superior compression and quality characteristics compared to their older JPEG and PNG counterparts. Encoding your images in these formats rather than JPEG or PNG means that they will load faster and consume less cellular data.”

Lighthouse

Before continuing, let's take a look at how Lighthouse (an open-source, automated tool for improving the performance, quality, and correctness of your web apps) can help us here. For best results, run Lighthouse in incognito mode. While in Chrome, run Control + Shift + N to open incognito mode. Then, go to your URL, open the developer tools, and select Audits.

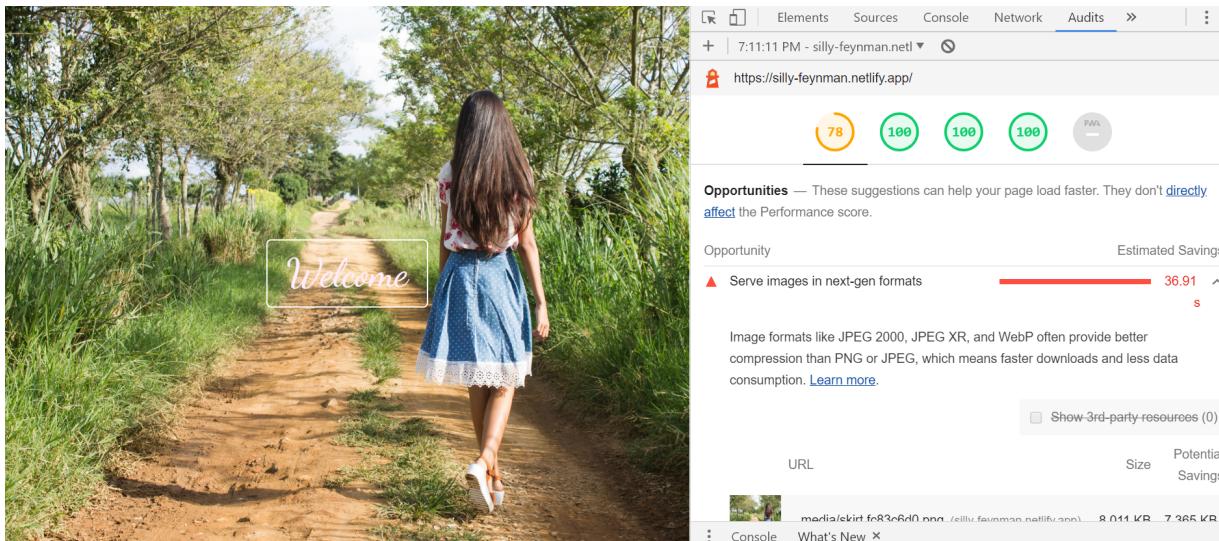


Click on “Generate report”.



The Problem

If you have a large image in either **.jpg** or **.png** format (the image in the screenshot is 2246 x 1473 pixels in size), it will take a long time to load. As you can see, Performance in Lighthouse is 78%, and it took 36.91 s to load. Lighthouse suggests serving images in next-gen formats.



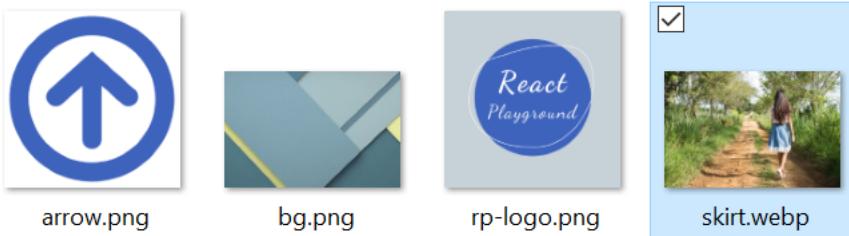
What To Do

1. First, convert your image from **.jpg/.png** to a next-gen format using an online converter like [this one](#). In this example, we're using this converter to convert from **.png** to **.webp**.

The screenshot shows the homepage of **ONLINE-CONVERT**. At the top, there's a banner with the text "Convert media free, fast and online. No software installation needed." and buttons for "Login" and "Register". Below the banner, the main navigation includes "Home", "File formats", "Blog", and "Developers". On the left, a sidebar titled "Converter" lists various conversion options under "Image converter", including "Convert To BMP", "Convert To EPS", "Convert To GIF", etc. The central content area features a section titled "Convert an image to WebP" with a "Drop Files here" button and a "Cloud" icon. To the right, there's a "Bookmark and share page" section with social sharing buttons for Facebook, Twitter, and LinkedIn, and a "Sponsored searches" section with links to "free image converter", "photo size converter", "convert jpg to png", "animated gif images", and "change photo to jpeg". A "More image converter" section at the bottom lists "1. Convert an image to the BMP format".

2. After converting the image into a **.webp** format, put the image into the appropriate folder. In this case, it has been placed in the **img** folder.

PC > Desktop > Web-Projects > react-contact-form > src > img



3. Import the image with the new **.webp** extension (line 3).

```
1 import React from 'react';
2 import { Container, Row, Col } from 'react-bootstrap';
3 import skirt from '../.../img/skirt.webp';
4
5 const Home = () => {
6   return (
7     -- REST OF CODE --
```

4. And that's it. Now it should work.



- The load time should now be faster. And when checking on Lighthouse, the Performance score has improved and is now at 90%. And we no longer get the message that suggests serving images in next-gen formats. And now when the pages on your app load, they should load faster.

Lighthouse Audit Results:

- Performance: 90
- Accessibility: 100
- Best Practices: 100
- SEO: 100

PWA Status: Progressive Web App

Metrics	
First Contentful Paint	2.9 s
Speed Index	3.5 s
Time to Interactive	2.9 s
First Meaningful Paint	2.9 s
First CPU Idle	2.9 s
Max Potential First Input Delay	190 ms