PerfPilot AI

Next.js Performance Made Simple

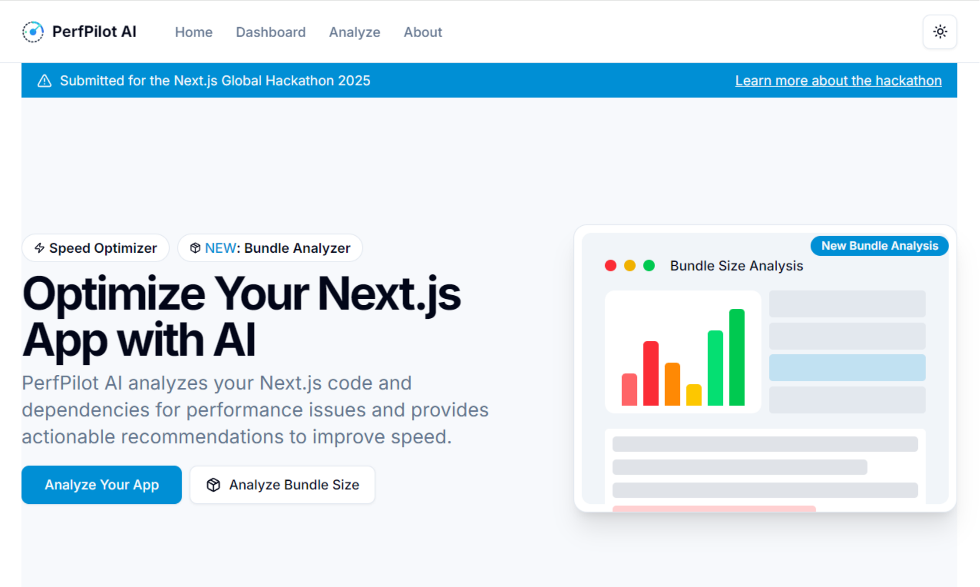
Next.js Hackathon Project

April 2025

The Performance Challenge

- 47% of users expect websites to load in under 2 seconds
- Next.js apps often suffer from:
  - Unoptimized images
  - Bloated JavaScript bundles
  - Inefficient data fetching
  - Missing Suspense boundaries
- Finding these issues requires hours of manual analysis

Introducing PerfPilot AI



Submitted for the Next.js Global Hackathon 2025

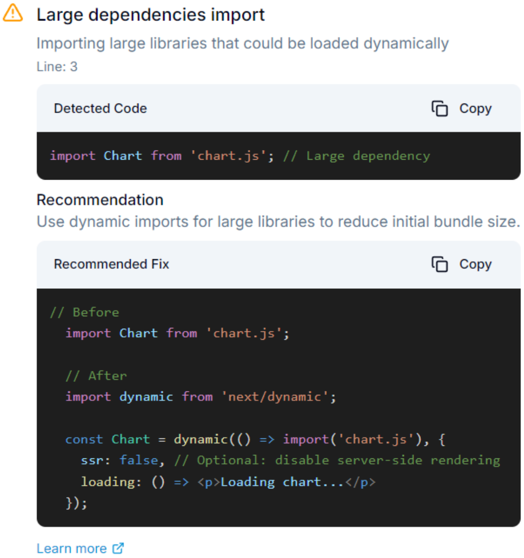
Learn more about the hackathon

Optimize Your Next.js App with AI

PerfPilot AI analyzes your Next.js code and dependencies for performance issues and provides actionable recommendations to improve speed.

1. Automatic Bottleneck Detection

- Scans component code and package.json
- Identifies performance anti-patterns
- Categorizes issues by severity
- Provides clear explanations of each issue



⚠ Large dependencies import

Importing large libraries that could be loaded dynamically

Line: 3

Detected Code

```
import Chart from 'chart.js'; // Large dependency
```

Recommendation

Use dynamic imports for large libraries to reduce initial bundle size.

Recommended Fix

```
// Before
  import Chart from 'chart.js';

  // After
  import dynamic from 'next/dynamic';

  const Chart = dynamic(() => import('chart.js'), {
    ssr: false, // Optional: disable server-side rendering
    loading: () => <p>Loading chart...</p>
  });
```

Learn more

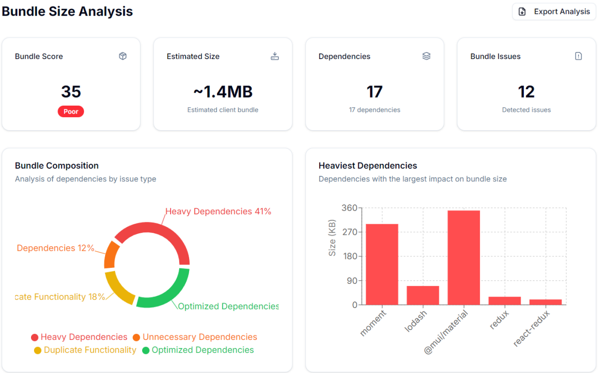
Example: Large Dependencies

Issue Detected: Large dependencies import
Severity: High
Line: 3

```
stlisting.2cmecommended Fix:stlisting
// After import dynamic from 'next/dynamic';
const Chart = dynamic(() => import('chart.js'),  ssr:  false, // Optional:
disable server-side rendering loading:  () => <p>Loading chart...</p> );
```

2. Bundle Size Analysis

- Analyzes your package.json
- Identifies heavy dependencies
- Detects duplicate packages
- Suggests lightweight alternatives
- Provides migration recommendations



Bundle Size Analysis

Export Analysis

Bundle Score
35
Poor

Estimated Size
~1.4MB
Estimated client bundle

Dependencies
17
17 dependencies

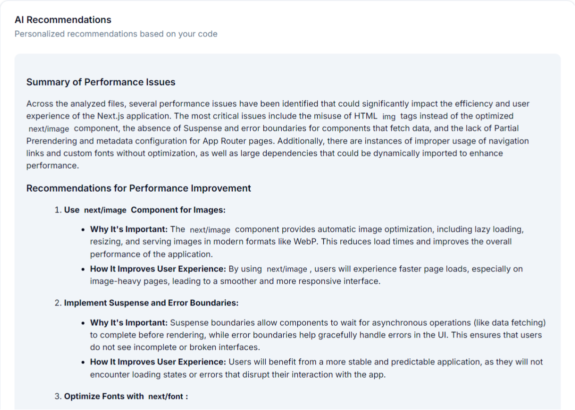
Bundle Issues
12
Detected issues

Bundle Composition
Analysis of dependencies by issue type

Heaviest Dependencies
Dependencies with the largest impact on bundle size

3. AI-Powered Recommendations

- Leverages Vercel AI SDK
- Generates code fixes with explanations
- Provides implementation guidance
- Contextualizes improvements
- Prioritizes for maximum impact



AI Recommendations

Personalized recommendations based on your code

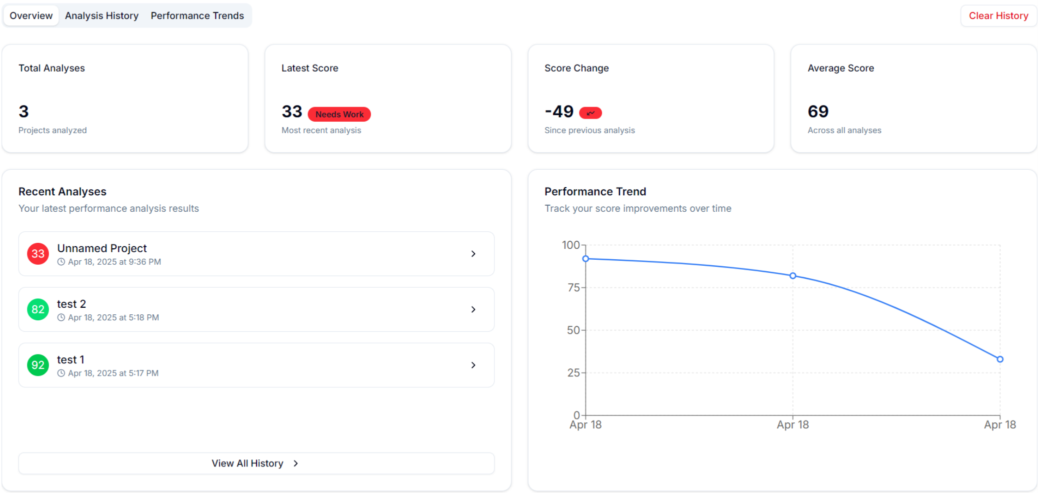
Summary of Performance Issues

Across the analyzed files, several performance issues have been identified that could significantly impact the efficiency and user experience of the Next.js application. The most critical issues include the misuse of HTML `img` tags instead of the optimized `next/image` component, the absence of Suspense and error boundaries for components that fetch data, and the lack of Partial Prerendering and metadata configuration for App Router pages. Additionally, there are instances of improper usage of navigation links and custom fonts without optimization, as well as large dependencies that could be dynamically imported to enhance performance.

Recommendations for Performance Improvement

1. **Use `next/image` Component for Images:**
   - **Why It's Important:** The `next/image` component provides automatic image optimization, including lazy loading, resizing, and serving images in modern formats like WebP. This reduces load times and improves the overall performance of the application.
   - **How It Improves User Experience:** By using `next/image`, users will experience faster page loads, especially on image-heavy pages, leading to a smoother and more responsive interface.
2. **Implement Suspense and Error Boundaries:**
   - **Why It's Important:** Suspense boundaries allow components to wait for asynchronous operations (like data fetching) to complete before rendering, while error boundaries help gracefully handle errors in the UI. This ensures that users do not see incomplete or broken interfaces.
   - **How It Improves User Experience:** Users will benefit from a more stable and predictable application, as they will not encounter loading states or errors that disrupt their interaction with the app.
3. **Optimize Fonts with `next/font`:**

4. Performance Tracking Dashboard



Total Analyses

3

Projects analyzed

Latest Score

33 Needs Work

Most recent analysis

Score Change

-49

Since previous analysis

Average Score

69

Across all analyses

Recent Analyses

Your latest performance analysis results

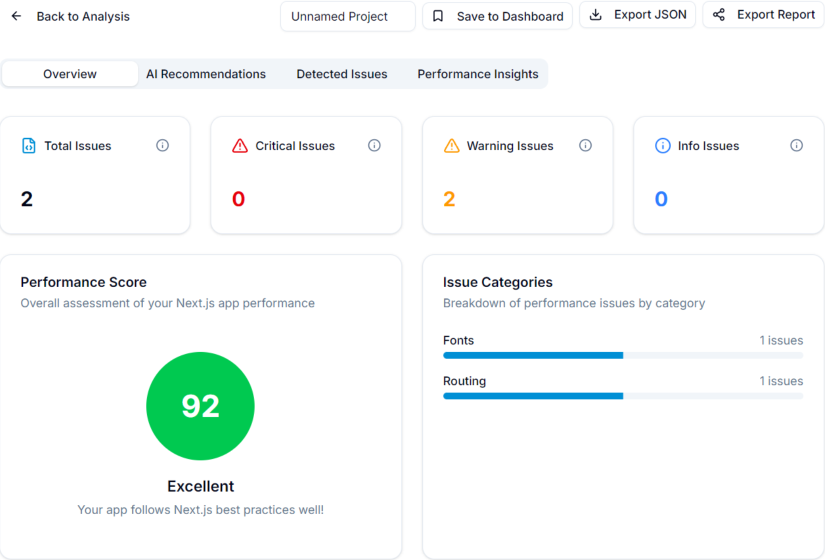
- 33 Unnamed Project — Apr 18, 2025 at 9:36 PM
- 82 test 2 — Apr 18, 2025 at 5:18 PM
- 92 test 1 — Apr 18, 2025 at 5:17 PM

Performance Trend

Track your score improvements over time

- Track improvements over time

Real Results



Total Issues 2

Critical Issues 0

Warning Issues 2

Info Issues 0

Performance Score

Overall assessment of your Next.js app performance

92

Excellent

Your app follows Next.js best practices well!

Issue Categories

Breakdown of performance issues by category

Fonts	1 issues
Routing	1 issues

Try PerfPilot AI Today

Thank You!

https://perfpilot.dev

github.com/belumume/perfpilot-ai

#NextJS #WebPerformance #AIinDevelopment