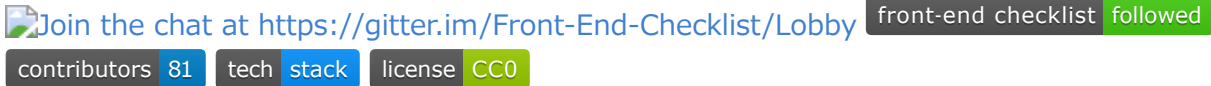


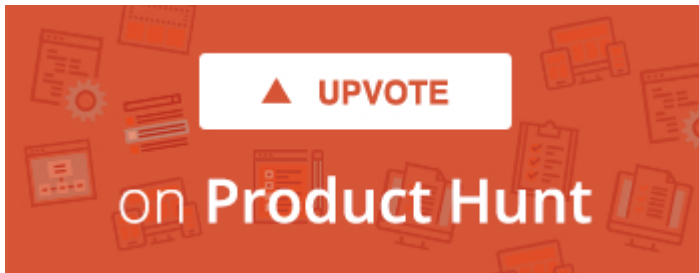
프론트엔드 체크리스트



프론트엔드 체크리스트는 당신의 HTML 사이트 또는 페이지를 프로덕션으로 런칭하기 이전에 가지고 있어야 할, 또한 테스트 되어야 할 전반적인 요소들의 집합입니다. 이 리스트는 다년간의 프론트엔드 개발자들의 경험으로 작성되었으며, 몇몇 타 오픈소스 체크리스트들의 참고를 통해 추가되었습니다.

목차

Product Hunt에 추천과 투표를 통해 프론트엔드 체크리스트를 널리 알리는데 도움을 주세요



1. **Head**
2. **HTML**
3. **웹폰트**
4. **CSS**
5. **이미지**
6. **자바스크립트**
7. **보안**
8. **성능**
9. **접근성**
10. **SEO**

이 리스트는 어떻게 사용하나요?

프론트엔드 체크리스트에 속해있는 모든 항목들은 대다수의 프로젝트에서 필요로 하는 사항들이지만, 몇몇 요소들은 생략되거나 필수적이 아닐 수도 있습니다(예를 들면 관리형 웹 어플리케이션의 경우 RSS 피드는 필요 없을 것입니다). 우리는 따라서 각각의 항목들을 3가지의 기준으로 구분하였습니다:

- **LOW**의 경우 해당 항목이 **권유** 되지만, 몇몇 특정한 상황에서는 생략될 수도 있습니다.
- **MEDIUM**의 경우 해당 항목이 **권고** 되지만, 굉장히 특정한 상황에서는 결국 생략이 될 수도 있습니다. 몇몇 요소의 경우, 생략 시 성능이나 SEO 측면에서 안 좋은 영향을 끼칠 수도 있습니다.
- **HIGH**의 경우 해당 항목은 어떠한 상황에서도 **생략될 수 없습니다**. 이를 생략하게 되면 당신의 페이지는 오동작하거나 접근, 또는 SEO에 문제가 발생할 것입니다. 이러한 요소들에 대해서 우선적으로 테스트 하시기 바랍니다.

몇몇 추가 내용들은 그것들이 어떠한 종류의 내용인지 이해하는데에 도움을 주기 위하여 이모티콘을 추가하였습니다. 이로 인해 체크리스트에서 해당 항목들을 이해하는 데에 도움이 될 것입니다:

- 📄: 문서 또는 기사
- 🛠: 온라인 도구 / 테스트 도구
- 📺: 미디어 또는 비디오 콘텐츠

Head

노트: [a list of everything](#) 에서 HTML 문서 내의 `<head>` 에서 사용할 수 있는 모든 것을 찾아보실 수 있습니다.

메타 태그

- ☐ **Doctype:** HIGH HTML5 을 사용하며, Doctype이 모든 HTML 페이지의 최상단에 위치함

```
<!-- Doctype HTML5 -->
<!doctype html>
```

📄 문자 인코딩 결정하기 - HTML5 W3C

다음 3개의 메타 태그(Charset, X-UA Compatible and Viewport)들은 다른 요소들에 비해 head 안에서 상단에 위치해야만 합니다.

- ☐ **Charset:** HIGH 문자집합(UTF-8)이 올바르게 선언됨

```
<!-- 이 문서에 대한 문자 인코딩을 설정 -->
<meta charset="utf-8">
```

- ☐ **X-UA-Compatible:** MEDIUM X-UA-Compatible 메타 태그가 존재함

```
<!-- Internet Explorer에게 최신의 렌더링 엔진을 사용하라고 지시 -->
<meta http-equiv="x-ua-compatible" content="ie=edge">
```

- 📄 Internet Explorer를 위한 레거시 문서 모드 지정하기

- ☐ **Viewport:** HIGH Viewport가 제대로 선언됨

```
<!-- 반응형 웹 디자인을 위한 Viewport 설정 -->
<meta name="viewport" content="width=device-width, initial-scale=1">
```

- ☐ **Title:** HIGH 모든 페이지에 title이 사용됨 (SEO 가이드: 웹사이트의 타이틀을 포함하여 65개 이하의 문자로 구성하기)

```
<!-- 문서의 Title -->
<title>55개 이하의 문자로 구성된 페이지 제목</title>
```

- [Title - HTML - MDN](#)
- [SERP Snippet Generator](#)
- ☐ **Description:** HIGH description이 제대로 기재됨 (설명값은 고유해야 하며, 150개 이하의 문자로 구성되어야 함)

```
<!-- Meta Description -->
<meta name="description" content="페이지에 대한 설명 (150개 이하의 문자)">
```

- [Meta Description - HTML - MDN](#)
- ☐ **Favicons:** MEDIUM 각각의 favicon이 제대로 생성되었고 올바르게 보여지는가? 만약 `favicon.ico` 파일만 가지고 있다면, 해당 내용을 페이지의 상단부에 추가하라. 일반적으로는 해당 태그를 사용할 필요는 없지만, 아래의 예시를 포함하는 것이 좋은 습관임. 오늘날에는 `.ico` 포맷보다 **PNG 포맷의 아이콘 사용을 추천함**(크기: 32x32px).

```
<!-- 표준 favicon -->
<link rel="icon" type="image/x-icon" href="https://example.com/favicon.ico">
<!-- 추천 favicon 포맷 -->
<link rel="icon" type="image/png" href="https://example.com/favicon.png">
```

- [Favicon Generator](#)
- [RealFaviconGenerator](#)
- [Favicon Cheat Sheet](#)
- [Favicons, Touch Icons, Tile Icons, etc. Which Do You Need? - CSS Tricks](#)
- [PNG favicons - caniuse](#)
- ☐ **Apple 터치 아이콘:** LOW 아이폰의 모바일 웹 어플리케이션의 아이콘으로 사용되는 favicon을 설정해주는 `apple-touch-icon` 속성을 사용함 (*최소한 200x200px의 크기로 Apple 아이콘을 생성하면, 필요한 모든 크기의 아이콘을 지원할 수 있음*)

```
<!-- Apple 터치 아이콘 -->
<link rel="apple-touch-icon" href="/custom-icon.png">
```

- [웹 어플리케이션 설정하기](#)
- ☐ **윈도우 타일:** LOW 윈도우 타일을 설정하는 `msapplication-config` 속성을 사용함

```
<!-- Microsoft 타일 -->
<meta name="msapplication-config" content="browserconfig.xml" />
```

browserconfig.xml 파일에서 사용되는 최소한의 XML 내용은 다음과 같습니다:

```
<?xml version="1.0" encoding="utf-8"?>
<browserconfig>
  <msapplication>
    <tile>
      <square70x70logo src="small.png"/>
      <square150x150logo src="medium.png"/>
      <wide310x150logo src="wide.png"/>
      <square310x310logo src="large.png"/>
    </tile>
  </msapplication>
</browserconfig>
```

- [📖 브라우저 설정 스키마 레퍼런스](#)

- ☐ **Canonical:** MEDIUM 콘텐츠의 중복을 피하기 위하여 `rel="canonical"` 을 사용함

```
<!-- 콘텐츠가 중복되는 문제를 방지하는데 도움이 됨 -->
<link rel="canonical" href="http://example.com/2017/09/a-new-article-to-red.html">
```

- [📖 표준 URL을 사용하기 - Search Console Help - Google Support](#)
- [📖 rel=canonical을 사용할 때 흔히 겪는 5가지 실수 - Google Webmaster Blog](#)

HTML 태그

- ☐ **언어 속성:** HIGH 현재 페이지 내의 언어에 알맞게 속성 값이 부여됨

```
<html lang="ko">
```

- ☐ **글자 방향 속성:** MEDIUM 글자들의 방향이 제대로 설정됨 (우리나라에서는 좌에서 우로 글씨를 읽고 쓰지만 몇몇 나라에서는 우에서 좌로 읽고 쓰는 경우도 있음)

```
<!-- rtl: right to left -->
<html dir="rtl">
```

- [📖 dir - HTML - MDN](#)

- ☐ **대체 언어:** LOW 현재 페이지를 언어에 맞게 대체할 수 있는 태그 속성 값을 사용함

```
<link rel="alternate" href="https://en.example.com/" hreflang="en">
```

- ☐ **조건부 주석:** LOW Internet Explorer 를 위한 조건부 주석을 사용함

-  [조건부 주석에 관하여 \(Internet Explorer\) - MSDN - Microsoft](#)

- ☐ **RSS 피드:** LOW 만일 페이지가 블로그이거나 기사가 있다면, RSS 링크에 대해 확인하시오
- ☐ **인라인 CSS Critical:** MEDIUM 페이지가 로딩되는 즉시(펼쳐지는 그 순간) 콘텐츠에 영향을 끼치는 CSS를 "critical CSS" 라고 함. 이는 당신의 실질적인 어플리케이션의 CSS 가 로딩되기 이전에 `<style></style>` 태그 사이에 최소화 된 상태로 한 줄로 추가되어 임베딩 됨

- ☐ [Critical by Addy Osmani on Github](#) 이 레포는 CSS Critical을 자동화 하는데에 도움을 줍니다.

- ☐ **CSS의 순서:** HIGH 모든 CSS 파일이 `<head>` 내에서 자바스크립트 파일보다 이전에 로딩이 완료됨 (자바스크립트 파일이 비동기적으로 로딩되는 특정한 경우는 제외함).



소셜미디어 관련 메타 태그

Facebook의 OG 와 **Twitter Cards** 를 사용하는 것은 어떠한 웹사이트든지간에 권고됩니다. 타 소셜미디어의 태그는 특정하게 그들을 대상으로 페이지를 노출 하려는 경우에는 고려해 볼 수 있습니다.

- ☐ **Facebook Open Graph:** LOW 모든 Facebook의 Open Graph (OG) 가 테스트 되었으며, 그것들 중에 누락된 정보나 잘못된 정보를 가지고 있지는 않나? (이미지의 경우 최소한 600 x 315 픽셀은 되어야 하며, 1200 x 630 픽셀 크기를 권장함)

Notes: Using `og:image:width` and `og:image:height` will specify the image dimensions to the crawler so that it can render the image immediately without having to asynchronously download and process it.

```
<meta property="og:type" content="website">
<meta property="og:url" content="https://example.com/page.html">
<meta property="og:title" content="Content Title">
<meta property="og:image" content="https://example.com/image.jpg">
<meta property="og:description" content="Description Here">
<meta property="og:site_name" content="Site Name">
<meta property="og:locale" content="en_US">
<!-- 다음의 태그는 필수는 아니지만, 포함하는 것을 추천함 -->
<meta property="og:image:width" content="1200">
<meta property="og:image:height" content="630">
```

-  [웹마스터를 위한 셰어링 가이드](#)
-  [셰어링 - 모범 사례](#)
- ☐ [Facebook OG testing](#) 도구를 이용하여 당신의 페이지 테스트하기

- ☐ **Twitter Card:** LOW

```
<meta name="twitter:card" content="summary">
<meta name="twitter:site" content="@site_account">
```

```
<meta name="twitter:creator" content="@individual_account">
<meta name="twitter:url" content="https://example.com/page.html">
<meta name="twitter:title" content="Content Title">
<meta name="twitter:description" content="Content description less than 200
characters">
<meta name="twitter:image" content="https://example.com/image.jpg">
```

- [📖 Twitter cards 시작하기 — Twitter Developers](#)
- [📖 Twitter card 검사기](#) 도구를 이용하여 당신의 페이지 테스트하기

↑ 목차로

HTML

모범 사례

- ☐ **HTML5 시맨틱 엘리먼트:** **HIGH** HTML5의 시맨틱 엘리먼트들이 적절히 사용됨 (header, section, footer, main... 등).

- [📖 HTML 레퍼런스](#)

- ☐ **에러 페이지:** **HIGH** 에러를 위한 404 페이지와 5xx 페이지가 존재하는가? 5xx 페이지는 서버로부터의 데이터를 전송받지 않고 독립적인 자체 CSS를 포함하고 있어야 함을 기억하라 (5xx 에러는 서버 에러이므로!).
- ☐ **Noopener:** **MEDIUM** 외부 링크를 `target="_blank"`를 이용하여 연 경우, **tab nabbing 피싱 공격**을 방지하기 위하여 `rel="noopener"` 속성을 사용해야만 한다. 만약 Firefox 옛 버전을 지원해야만 한다면, `rel="noopener noreferrer"`을 사용하라.

- [📖 rel=noopener에 대하여](#)

- ☐ **주석 지우기:** **LOW** 웹사이트를 프로덕션 하기 이전에 불필요한 코드는 제거하였는지, 주석은 제거하였는지 점검하라

HTML testing

- ☐ **W3C 규격:** **HIGH** 모든 페이지는 HTML 코드 내에서 일어날 수 있는 경우를 확인하기 위하여 W3C의 validator를 이용하여 테스트 되어야 함

- [📖 W3C 검사기](#)

- ☐ **HTML Lint:** **HIGH** 도구를 이용하여 HTML 코드 내에 발생할 수 있는 문제들을 분석하도록 하자

- [📖 Dirty markup:](#) HTML 코드를 정돈해주는 온라인 도구

- [📖 Sonar a linting tool for the web](#)

- ☐ **Link checker:** **HIGH** 페이지 내에 깨진 링크는 없는지, 404 에러가 존재하지 않는지 다시 한번 확인하도록 함

- [📖 W3C Link Checker](#)

- ☐ **광고차단기 테스트:** MEDIUM 광고차단기가 활성화 된 상태에서도 콘텐츠가 제대로 보여짐 (사람들에게 광고차단기를 비활성화 해달라고 메시지를 알릴수도 있습니다)

↑ 목차로

웹폰트

노트: 웹폰트를 사용하면 스타일링 되지 않은 글자나 보이지 않는 글자들이 깜빡일 수 있습니다. 실패했을 경우를 대비한 대체용 폰트를 포함하거나, 웹폰트 로더를 활용하여 이러한 동작들을 제어하세요.

- ☐ **구글의 웹폰트에 대한 기술적 고려사항**
- ☐ **웹폰트 포맷:** HIGH WOFF, WOFF2 와 TTF는 모든 최신 브라우저에서 지원됨
 - ☐ [WOFF - Web Open Font Format - Caniuse.](#)
 - ☐ [WOFF 2.0 - Web Open Font Format - Caniuse.](#)
 - ☐ [TTF/OTF - TrueType and OpenType font support](#)
 - ☐ [Using @font-face - CSS-Tricks](#)
- ☐ **웹폰트 크기:** HIGH 모든 종류(이탤릭, 볼드체 등등)를 포함한 웹폰트 크기의 총 합계는 2 MB를 넘지 않도록 함
- ☐ **웹폰트 로더:** LOW 웹폰트 로더를 이용하여 폰트가 로딩되는 동작을 제어하시오
 - ☐ [Typekit 웹폰트 로더](#)

↑ 목차로

CSS

노트: 대다수의 프론트엔드 개발자들이 따르는 [CSS 가이드라인](#)과 [Sass 가이드라인](#) 을 살펴보세요. 만약 모르는 CSS 속성 값이 있다면, [CSS 레퍼런스](#)를 참조하길 바랍니다. 또한 CSS의 일관성을 위한 [코드 가이드](#)도 읽어보기 바랍니다.

- ☐ **반응형 웹 디자인:** HIGH 웹사이트가 반응형으로 디자인 됨
- ☐ **CSS Print:** MEDIUM 프린터가 사용할 print 스타일시트 값이 설정되었고, 각각의 페이지마다 올바르게 설정됨
- ☐ **CSS 전처리기:** MEDIUM 디자인에 CSS 전처리기를 이용함 (추천: [Sass](#)).
- ☐ **고유 ID값:** HIGH 여러 개의 ID 값이 사용된 경우, 각각의 ID 값은 페이지 내에 고유해야함
- ☐ **Reset CSS:** HIGH 최신의 Reset CSS (reset, normalize나 reboot) 이 사용됨 (*Bootstrap 이/나 Foundation 같은 CSS 프레임워크를 사용할 경우, Normalize가 이미 포함되어 있음*)
 - ☐ [Reset.css](#)
 - ☐ [Normalize.css](#)
 - ☐ [Reboot](#)
- ☐ **JS 접두사:** LOW js-로 시작하는 자바스크립트 파일 내에서 사용되는 모든 클래스나 ID는 CSS 파일에서 스타일링 되지 않도록 함

```
<div id="js-slider" class="my-slider">
<!-- 또는 -->
<div id="id-used-by-cms" class="js-slider my-slider">
```

- ☐ **CSS 임베딩 또는 인라인:** **HIGH** 어떠한 경우에도 CSS를 직접 임베딩하거나 인라인으로 사용하지 마시오! 타당한 이유가 있는 경우에만 사용하시오 (예: 슬라이더 내의 background-image, 또는 CSS critical)
- ☐ **벤더 프리픽스:** **HIGH** CSS 벤더 프리픽스들이 사용되었고 브라우저 지원 호환성에 따라 알맞게 생성되었는지 확인

- ☐ [Autoprefixer CSS online](#)

성능

- ☐ **파일 단일화:** **HIGH** CSS 파일들이 하나의 CSS 파일로 단일화 되었음 (*HTTP/2의 경우는 여러 개의 파일을 사용하는 것이 더 성능에 좋음*)
- ☐ **최소화:** **HIGH** 모든 CSS 파일들이 최소화 됨
- ☐ **Non-blocking:** **MEDIUM** CSS 파일들은 DOM이 로딩하는데에 방해가 되지 않도록 비동기적으로 로드되어야 함

- ☐ [loadCSS by filament group](#)
- ☐ [Example of preload CSS using loadCSS](#)

- ☐ **사용하지 않은 CSS:** **LOW** 사용되지 않은 CSS는 제거함

- ☐ [UnCSS Online](#) ☐
- ☐ [PurifyCSS](#)
- ☐ [Chrome DevTools Coverage](#)

CSS 테스트

- ☐ **Stylelint:** **HIGH** 모든 CSS와 SCSS 파일들에 아무런 에러가 없는지 확인

- ☐ [stylelint, a CSS 린터](#)
- ☐ [Sass 가이드라인](#)



- ☐ **반응형 웹 디자인:** **HIGH** 모든 페이지가 다음 지점에서 테스트 완료되었음: 320px, 768px, 1024px (그 외 당신이 필요한 크기에 따라 다를 수 있음)
- ☐ **CSS 검사기:** **MEDIUM** CSS가 제대로 테스트 되었고, 오류들이 알맞게 수정되었음

- ☐ [CSS 검사기](#)

- ☐ **데스크탑 브라우저:** **HIGH** 모든 페이지가 모든 현존하는 데스크탑 브라우저에서 테스트 됨 (Safari, Firefox, Chrome, Internet Explorer, EDGE... 등).
- ☐ **모바일 브라우저:** **HIGH** 모든 페이지가 모든 현존하는 모바일 브라우저에서 테스트 됨 (Native browser, Chrome, Safari... 등).
- ☐ **운영체제:** **HIGH** 모든 페이지가 모든 현존하는 운영체제에서 테스트 됨 (Windows, Android, iOS, Mac... 등).

- ☐ **Pixel perfect:** HIGH 페이지가 Pixel perfect한 상태(원래 의도했던 디자인대로 화면에 보여짐)인가? 창작물에 따라서 100% 정확하지 않을 수도 있지만, 의도했던 템플릿에 가까워야 함

Pixel Perfect - Chrome 확장도구



- ☐ **글자 방향:** HIGH 다국어를 지원해야 할 경우, 글자 방향에 맞게 모든 페이지가 정상 동작함 (LTR / RTL)
 -  [Building RTL-Aware Web Apps & Websites: Part 1 - Mozilla Hacks](#)
 -  [Building RTL-Aware Web Apps & Websites: Part 2 - Mozilla Hacks](#)

↑ 목차로

이미지

노트: 이미지 최적화의 완전한 이해를 위해서는, Addy Osmani가 쓴 무료 ebook [Essential Image Optimization](#) 을 확인하세요.

모범 사례

- ☐ **최적화:** HIGH 모든 이미지가 브라우저에 렌더링 될 수 있도록 최적화 되었나? 홈페이지 같이 성능이 중요한 페이지에는 WebP 포맷을 사용할 수도 있음
 - ☐ [Imagemin](#)
 - ☐ [ImageOptim](#)를 사용하여 당신의 이미지를 무료로 최적화하세요
 - ☐ [Kraken.io](#)를 사용하여 깔끔한 png 이미지와 jpg 이미지의 최적화를 수행할 수 있습니다. 파일당 1MB의 크기에 대해서는 무료입니다
- ☐ **Picture/Srcset:** MEDIUM You use picture/srcset to provide the most appropriate image for the current viewport of the user.
 -  [srcset을 사용하여 반응형 이미지 만드는 방법](#)
- ☐ **레티나 디스플레이 지원:** LOW 레티나 디스플레이를 지원하기 위하여 당신의 현 레이아웃에 해당하는 2배, 또는 3배 이상 큰 이미지를 지원함
- ☐ **이미지 스프라이트:** MEDIUM 작은 이미지의 경우 스프라이트 파일로 구성되어 있음 (아이콘의 경우, SVG 스프라이트 이미지 일 수도 있음).
- ☐ **너비와 높이:** HIGH 모든 `` 태그에 너비와 높이가 설정되었음 (px이나 %로 지정하지 마시오)
- ☐ **대체 텍스트:** HIGH 모든 `` 태그가 이미지를 잘 서술하는 대체 텍스트를 가지고 있음 (`alt` 속성으로 부여)
-  [대체 텍스트: 최고의 가이드](#)
- ☐ **Lazy 로딩:** MEDIUM 이미지들이 lazy 로드 됨 (자바스크립트 미지원에 대한 예외처리가 항상 제공되어야 함)

↑ 목차로

자바스크립트

모범 사례

- ☐ **인라인 자바스크립트:** **HIGH** HTML 코드와 섞어 자바스크립트 코드를 인라인으로 포함하지 않도록 하시오
- ☐ **파일 단일화:** **HIGH** 자바스크립트 파일들이 하나의 자바스크립트 파일로 단일화 되었음
- ☐ **최소화:** **HIGH** 모든 자바스크립트 파일이 최소화 되었음(뒤에 **.min** 접미사를 붙이는 것을 추천)

• [리소스\(HTML, CSS, and JavaScript\) 최소화하기](#)

- ☐ **자바스크립트 보안:**

• [자바스크립트를 활용하여 보안에 안전한 어플리케이션 개발 가이드라인*](#)

- ☐ **Non-blocking:** **MEDIUM** JavaScript 파일들은 **async**와 **defer** 속성값을 이용하여 비동기적으로 로드 되어야 함

• ☐ [렌더링을 막는 자바스크립트 제거하기](#)

- ☐ **Modernizr:** **LOW** 특정한 기능을 지칭하고 싶다면, 커스터마이징 된 Modernizr를 이용하여 **<html>** 태그 내에 클래스들을 추가할 수 있음

• ☐ [당신의 Modernizr 커스터마이징 하기](#)

자바스크립트 테스트

- ☐ **ESLint:** **HIGH** 표준 규칙이나 당신의 설정에 따라 ESLint를 에러 없이 통과함

• ☐ [ESLint - 자바스크립트와 JSX를 위한, 언제든지 사용할 수 있는 린팅 유틸리티](#)

↑ 목차로

보안

당신의 웹사이트를 미리 검토하고 확인하세요

- [securityheaders.io](#)
- [Observatory by Mozilla](#)
- [ASaWeb - Automated Security Analyser for ASP.NET Websites](#)

모범 사례

- ☐ **HTTPS:** **MEDIUM** 페이지 내에 존재하는 모든 외부 콘텐츠(플러그인, 이미지...)에 대해서도 HTTPS가 사용되었음.

- ☐ [Let's Encrypt - Free SSL/TLS Certificates](#)
- ☐ [Free SSL Server Test](#)
- ☐ [Strict Transport Security](#)

- ☐ **HTTP Strict Transport Security (HSTS):** **MEDIUM** HTTP 헤더 값으로 'Strict-Transport-Security'가 설정됨.

- ☐ [Check HSTS preload status and eligibility](#)
- ☒ [HSTS 참조서 - OWASP](#)
- ☒ [전송 계층 보안 참조서 - OWASP](#)

- ☐ **사이트 간 요청 위조(CSRF; Cross Site Request Forgery):** **HIGH** CSRF 공격을 방지하기 위하여 위하여 당신의 서버 쪽으로 발생하는 모든 HTTP 요청이 합법적이고 당신의 웹사이트나 어플리케이션으로부터 발생한 것임을 확신함

- ☒ [CSRF 예방 참조서 - OWASP](#)

- ☐ **사이트 간 스크립팅(XSS; Cross Site Scripting):** **HIGH** 당신의 페이지나 웹사이트가 사이트 간 스크립팅이 발생할 여지가 전혀 없음

- ☒ [XSS 예방 참조서 - OWASP](#)
- ☒ [DOM 기반 XSS 예방 참조서 - OWASP](#)

- ☐ **Content Type Options** **MEDIUM** Prevents Google Chrome and Internet Explorer from trying to mime-sniff the content-type of a response away from the one being declared by the server.

- ☒ [X-Content-Type-Options - Scott Helme](#)

- ☐ **X-Frame-Options (XFO)** **MEDIUM** 방문자를 클릭재킹 공격으로부터 보호함

- ☒ [X-Frame-Options - Scott Helme](#)
- ☒ [RFC7034 - HTTP Header Field X-Frame-Options](#)

- ☐ **컨텐츠 보안 정책(CSP; Content Security Policy)** **MEDIUM** 사이트 내의 컨텐츠가 어떻게 로딩되고, 어디서 로딩되도록 허가받았는지를 확인. 이를 적용함으로써 클릭재킹 공격을 방지할 수 있음

- ☒ [컨텐츠 보안 정책 소개 - Scott Helme](#)
- ☒ [컨텐츠 보안 정책 가이드라인 - Scott Helme](#)
- ☒ [컨텐츠 보안 정책 가이드라인 - OWASP](#)

↑ 목차로

성능

모범 사례

- ☐ **페이지 용량:** **HIGH** 각 페이지의 용량이 500KB 이하임

- ☐ [웹사이트 페이지 분석](#)
- ☒ [페이지 용량을 제한함으로써 웹을 더욱 가볍게 만드세요](#)

- ☐ **최소화:** **MEDIUM** HTML이 최소화가 되었음

- ☐ [W3C 검사기](#)

- ☐ **Lazy 로딩:** **MEDIUM** 이미지, 스크립트, CSS 파일들이 lazy 로드 되어서 현 페이지의 응답시간을 향상시킴 (각 섹션의 자세한 부분을 참조하십시오).

- ☐ **쿠키 크기:** 쿠키를 사용한다면, 각 쿠키의 크기가 4096 바이트를 넘지 않고, 도메인 내에 20개 이상의 쿠키를 가지지 않도록 주의하시오

- ☐ 쿠키 사양: RFC 6265
- ☐ 쿠키
- ☐ 브라우저 쿠키의 제한점

- ☐ **서드 파티 컴포넌트:** **MEDIUM** 공유하기 버튼처럼 외부 자바스크립트 파일에 의존하는 서드파티 iframe이나 컴포넌트의 경우, 가급적 정적인 컴포넌트로 교체하여서 외부 API 호출을 제한하고 사용자들의 활동들을 외부로 유출되지 않도록 하시오

- ☐ 간단한 공유하기 버튼 생성기

다음에 발생할 HTTP 요청을 미리 준비하기

- ☐ [Explanation of the following techniques](#)

- ☐ **DNS resolution:** **LOW** DNS of third-party services that may be needed are resolved in advance during idle time using **dns-prefetch**.

```
<link rel="dns-prefetch" href="https://example.com">
```

- ☐ **Preconnection:** **LOW** DNS lookup, TCP handshake and TLS negotiation with services that will be needed soon is done in advance during idle time using **preconnect**.

```
<link rel="preconnect" href="https://example.com">
```

- ☐ **Prefetching:** **LOW** Resources that will be needed soon (e.g. lazy loaded images) are requested in advance during idle time using **prefetch**.

```
<link rel="prefetch" href="image.png">
```

- ☐ **Preloading:** **LOW** Resources needed in the current page (e.g. scripts placed at the end of **<body>**) in advance using **preload**.

```
<link rel="preload" href="app.js">
```

- ☐ [prefetch와 preload의 차이점](#)

성능 테스트

- ☐ **Google PageSpeed:** **HIGH** 홈페이지 뿐 아니라 모든 페이지가 테스트 완료 되었고 최소한 100점 만점 90점은 획득하였음

- ☐ [Google PageSpeed](#)
- ☐ [Google에서 모바일 속도를 측정해보세요](#)
- ☐ [WebPagetest - 웹사이트 성능 및 최적화 테스트](#)
- ☐ [GTmetrix - 웹사이트 속도 및 성능 최적화](#)

↑ 목차로

접근성

노트: 유튜브의 재생 목록을 확인해보세요 [A11ycasts with Rob Dodson](#) 📺

모범 사례

- ☐ **Progressive enhancement:** MEDIUM 메인 네비게이션이나 검색과 같은 대다수의 기능들이 자바스크립트가 작동하지 않고도 동작해야 함
- ☐ [Chrome 개발자 도구를 이용하여 자바스크립트 키고 끄기](#)
- ☐ **Color contrast:** MEDIUM Color contrast should at least pass WCAG AA (AAA for mobile).
- ☐ [Contrast ratio](#)

헤딩

- ☐ **H1:** HIGH 모든 페이지 내에 웹사이트의 타이틀과 다른 H1 태그가 존재해야 함
- ☐ **헤딩:** HIGH 헤딩이 올바른 순서(H1부터 H6까지)로 적절히 사용되어야 함
- 📺 [헤딩과 랜드마크가 그렇게 중요한 이유 -- A11ycasts #18](#)

랜드마크

- ☐ **banner 역할:** HIGH `<header>` 태그가 `role="banner"` 속성값을 가지고 있음
- ☐ **navigation 역할:** HIGH `<nav>` 태그가 `role="navigation"` 속성값을 가지고 있음
- ☐ **main 역할:** HIGH `<main>` 태그가 `role="main"` 속성값을 가지고 있음
- ☐ [ARIA 랜드마크를 이용하여 페이지 내의 특정 지점을 식별하기](#)
- ☐ [ARIA 역할의 범주](#)

시맨틱

- ☐ **특정한 HTML5의 input 타입들의 사용:** MEDIUM 이 항목은 각각 다른 input 타입에 대하여 개별적인 키패드나 위젯을 보여주는 모바일 장치들에 대해 특히 더욱 중요함
- ☐ [모바일 Input 타입](#)

폼

- ☐ **레이블:** HIGH 레이블은 각각의 입력 폼 엘리먼트와 연관됨. 레이블이 보여질 수 없는 경우 `aria-label` 을 대신 사용하시오
- ☐ [aria-label 속성 사용하기 - MDN](#)

접근성 테스트

- ☐ **접근성 표준 테스트:** **HIGH** WAVE 도구를 이용하여 당신의 페이지가 접근성 표준을 만족하였는지 테스트 해보세요

- ☐ [Wave 테스트](#)

- ☐ **키보드 네비게이션:** **HIGH** 키보드만을 이용하여 웹사이트를 예측 가능하도록 움직일 수 있는지 테스트 하시오. 모든 인터랙티브 엘리먼트들은 접근 가능하고 사용 가능해야 함
- ☐ **스크린 리더:** **MEDIUM** 모든 페이지들이 스크린 리더 (VoiceOver, ChromeVox, NVDA or Lynx) 테스트를 완료함
- ☐ **포커스 스타일링:** **HIGH** 포커스 되지 않은 경우, 눈에 보이는 상태의 CSS로 대체되어야 함

- ☐ [포커스 관리하기 - A11ycasts #22](#)

↑ 목차로

SEO

- ☐ **구글 애널리틱스:** **HIGH** 구글 애널리틱스가 설치되었고 제대로 설정되었음
- ☐ **적절한 제목 배치:** **MEDIUM** 제목은 현 페이지의 내용을 이해하는 데에 도움을 줌
- ☐ **sitemap.xml:** **HIGH** sitemap.xml 파일이 존재하고 Google Search Console(예전 이름: Google Webmaster Tools)으로 제출되었음
- ☐ **robots.txt:** **HIGH** robots.txt 파일이 웹페이지를 블록킹 하지 않음

- ☐ [Google Robots 테스트 도구](#)를 이용하여 당신의 robots.txt 파일을 테스트 해보세요

- ☐ **구조화 된 데이터:** **HIGH** 구조화 된 데이터를 사용하여 페이지가 테스트되었고 에러가 존재하지 않는가? 구조화 된 데이터는 웹 크롤러가 현 페이지 내의 콘텐츠를 이해하는 데에 도움이 됨

- ☐ [구조화 된 데이터 소개 - Search - Google Developers](#)
- ☐ [구조화 된 데이터 테스트 도구](#)를 이용하여 당신의 페이지를 테스트 해보세요
- ☐ 구조화 된 데이터로 사용될 수 있는 단어들의 목록을 만들어보세요 [Schema.org Full Heirarchy](#)

- ☐ **HTML 사이트맵:** **MEDIUM** HTML 사이트맵이 제공되었으며 웹사이트의 푸터 내에 존재하는 링크를 통하여 접근이 가능함

- ☐ [사이트맵 가이드라인 - Google Support](#)
- ☐ [Sitemap 생성기](#)

↑ 목차로

번역

프론트엔드 체크리스트는 다른 언어로도 이용 가능합니다. 모든 번역자들과 그들의 멋진 노력에 감사합니다!

- JP Japanese: [miya0001/Front-End-Checklist](#)
- ES Spanish: [eoasakura/Front-End-Checklist-ES](#)

- CN Chinese: [JohnsenZhou/Front-End-Checklist](#)
- KR Korean: [kesuskim/Front-End-Checklist](#)
- BR Portuguese: [jcezarms/Front-End-Checklist](#)
- VN Vietnamese: [euclid1990/Front-End-Checklist](#)
- TW Traditional Chinese: [EngineLin/Front-End-Checklist](#)

프론트엔드 체크리스트 배지

만약 당신이 프론트엔드 체크리스트의 규칙을 따르고 있다고 보여주길 원한다면, 하단의 배지를 README 파일에 추가하세요!

→ 

```
[![Front-End_Checklist followed](https://img.shields.io/badge/Front-End_Checklist-followed-brightgreen.svg)](https://github.com/thedaviddias/Front-End-Checklist/)
```

↑ 목차로

프로젝트에 기여

이슈를 새로 생성하거나 PR을 날려서 수정 사항이나 추가할 부분에 대해 알려주세요.

가이드

프론트엔드 체크리스트 레포지토리는 2개의 브랜치로 구성되어져 있습니다:

1. master

이 브랜치는 [프론트엔드 체크리스트](#) 웹사이트에 반영되는 `README.md`파일로 구성되어져 있습니다.

2. develop

이 브랜치는 필요하다면 구조나 콘텐츠에 상당한 변화를 줄 수 있습니다. 간단한 에러 수정을 하거나 새로운 항목을 추가할 경우, master 브랜치에 직접 하는 것을 추천드립니다.

Contributors

Check out all the super awesome [contributors](#).

Support

질문이나 제안이 있다면, 주저하지 말고 Gitter나 Twitter를 이용하세요:

- [Chat on Gitter](#)
- [Facebook](#)
- [Twitter](#)

저자

[David Dias](#)

라이선스



[↑ 목차로](#)