

**Search Engine Optimized Blog**

***SEO BLOG***



# Spis treści

[Spis treści 1](#_Toc499009052)

[Overview 1](#_Toc499009053)

[Features 1](#_Toc499009054)

[Client side 1](#_Toc499009055)

[Search angine 2](#_Toc499009056)

[Admin Panel 3](#_Toc499009057)

# Overview

This application should allow to create high search engine optimized blog.

# Features

--(server) - means that this features is provided by server side only

--#1, #2, #3, ... - means feature priority

## Client side

1. Contact form #1
2. Reviews and/or comments with external identify (google, Facebook) #2
3. Social Media buttons (likes, twits,...) #2
4. Desktop/Tablet/Mobile view #1
5. Content elements:
   1. Logo #1
   2. Navigation Menu #1 (Menus: top, main, left)#3
   3. Slider #1
   4. Headings #1
   5. Paragraphs #1
   6. Images #1
   7. Videos #1
   8. Special blocks:
      1. Code block (c#, js, Html) #3
      2. draw.io viewer #4

## Search engine

1. Accelerated Mobile Pages (AMP) #1

This technology allows google to cache pages in their cache its speed up page loading from google search. If page support this technology than page is higher in search results

**Implementation:** <https://www.ampproject.org/>

1. Structured data: #1
   1. Website
   2. Webpage
   3. Organization
   4. Article
   5. Breadcrumbs
   6. Site navigation elements

**Implementation:** <http://schema.org/> and <https://developers.google.com/search/docs/guides/search-features>

1. Gzip compression (server) #1

**Implementation:** <https://docs.microsoft.com/en-us/iis/configuration/system.webserver/httpcompression/>

1. Keep-alive attribute (server) #1

**Implementation:** <https://docs.microsoft.com/en-us/iis/configuration/system.webserver/httpprotocol/>

1. CDN (server) #4
2. Optimized images (server) #1

**Implementation:** Use tiny png Api for .Net. Each blog instances should have own TinyPNG Api Key, it should be configured by user in admin panel. <https://tinypng.com/developers>

1. Google Analytics (server) #1
2. Google webmaster tools (server) #1
3. Google Tag Manager (server) #2
4. Robots.txt (server) #1
5. H1,H2,H3,H4,strong, #1
6. Friendly URLs #1
7. Image Alts #1
8. No inline CSS #1
9. Data URIs for top page images (logo especially) (server) #2
10. Server Cache (server) #1
11. Expires headers (browser cache) (server) #1
12. JS/CSS magnification (server) #1
13. URL canonicalization www to non www (server) #1
14. Https (server) #1
15. No server signature (server) #2
16. Email protection #2
17. Top page style including to page code #2
18. Meta tags:
    1. Hreflang (server) #1
    2. Description (server) #1
    3. Charset (server) #1
    4. Robots (server) #2
    5. Social media meta tags: #1
       1. Facebook
       2. Twitter
       3. Pinterest
19. Links:
    1. Canonical (server) #2
    2. Pagination (prev, next) (server) #2
20. Title tag (server) #1
21. Site map (server) #1
22. RWD #1
23. Only required styles loading #2
24. Only required JS loading #3

## Admin Panel

1. Working copy/copies / Publications #1
2. Versions history? #2
3. Allow users to edit texts just in end view html #1
4. Allow to modify element properties in properties block (right side of window) #1:
   1. All elements
      1. Alignment
      2. Width
   2. Text
      1. Colour
      2. Decoration
   3. Image
      1. Alt
      2. Url (inline/block)
   4. Link
      1. Url
      2. Text (inline)
      3. Target
   5. Gogle Map (iframe)
      1. Url
5. Text properties should use class styling (no inline styles and additional tags) #1
6. Allow users to build his website structure in tree. (notify user if tree have more than 3 levels) #1
7. Allow user to use tags for his articles
8. Allow user to create series of blogs
9. Allow user to add new element to page #1
10. Allow user to move elements #1
11. Desktop View edition with Mobile View Block(Preview/Editon) #3
    1. User can choose device from list
    2. Set custom width, height
    3. Horizontal/Vertical