

Kompletny Plan: Strona z Ankietą + Pozycjonowanie SEO

Spis treści

1. [Część 1: Tworzenie i Hosting Strony](#)
2. [Część 2: Pozycjonowanie SEO](#)
3. [Harmonogram i Budżet](#)

Część 1: Tworzenie i Hosting Strony

Krok 1: Wybór technologii

Opcja A: Bez kodowania (najszybsza)

Narzędzie	Koszt	Czas wdrożenia
Google Forms + Google Sites	Darmowe	1-2 godziny
Tally.so + Carrd.co	Darmowe	2-4 godziny
Typeform + Webflow	0-50 zł/mies	1 dzień

Opcja B: Z kodowaniem (rekомendowana dla SEO)

Stack	Zalety
Next.js + Vercel	SSR/SSG, szybkie, darmowy hosting
Nuxt.js + Netlify	Vue.js, dobry dla SEO
HTML/CSS/JS + GitHub Pages	Najprostsze, darmowe

Krok 2: Rejestracja domeny

Gdzie kupić domenę

- OVH.pl - od 29 zł/rok (.pl)
- home.pl - od 39 zł/rok
- Cloudflare Registrar - najtaniej, bez marży

Wybór nazwy domeny

☒ Dobre przykłady:
- ubezpieczenia-porownaj.pl
- polisa-na-zycie.pl
- ubezpiecz-rodzine.pl

☒ Unikaj:
- bardzo długich nazw
- myślników więcej niż 1
- trudnych do wymówienia

Krok 3: Struktura strony

```
/  
|__ index.html          # Strona główna z ankietą  
|__ /ubezpieczenia-na-zycie/    # Główna fraza SEO  
|__ /porownanie/           # Podstrona  
|__ /kalkulator/            # Narzędzie  
|__ /blog/                  # Artykuły (ważne dla SEO!)  
|   |__ najtansze-ubezpieczenie-2024/  
|   |__ jak-wybrac-polise/  
|   |__ ubezpieczenie-dla-rodziny/  
|__ sitemap.xml  
|__ robots.txt  
|__ favicon.ico
```

Krok 4: Implementacja strony

4.1 Podstawowa struktura HTML

```

<!DOCTYPE html>
<html lang="pl">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <!-- SEO Meta Tags -->
    <title>Ubezpieczenia na życie - Porównaj oferty | NazwaStrony</title>
    <meta name="description" content="Porównaj ubezpieczenia na życie online.
        Sprawdź najlepsze oferty PZU, Allianz, Warta. Bezpłatna ankieta i wycena.">
    <meta name="keywords" content="ubezpieczenia na życie, polisa na życie,
        ubezpieczenie rodzinne">

    <!-- Open Graph (social media) -->
    <meta property="og:title" content="Ubezpieczenia na życie - Porównaj">
    <meta property="og:description" content="Porównaj oferty ubezpieczeń...">
    <meta property="og:image" content="https://strona.pl/og-image.jpg">
    <meta property="og:url" content="https://strona.pl">

    <!-- Canonical URL -->
    <link rel="canonical" href="https://strona.pl/ubezpieczenia-na-zycie">

    <link rel="stylesheet" href="style.css">
</head>
<body>
    <header>
        <nav>
            <a href="/">Start</a>
            <a href="/ubezpieczenia-na-zycie">Ubezpieczenia</a>
            <a href="/blog">Blog</a>
        </nav>
    </header>

    <main>
        <h1>Ubezpieczenia na życie - znajdź najlepszą ofertę</h1>

        <!-- Ankieta -->
        <section id="ankieta">
            <h2>Wypełnij ankietę i otrzymaj bezpłatną wycenę</h2>
            <form id="insurance-form">
                <!-- pola formularza -->
            </form>
        </section>
    </main>

    <!-- Structured Data -->
    <script type="application/ld+json">
    {
        "@context": "https://schema.org",
        "@type": "WebPage",
        "name": "Ubezpieczenia na życie",
        "description": "Porównaj ubezpieczenia na życie online"
    }
</script>
</body>
</html>

```

4.2 Formularz ankiety (JavaScript)

```
// form.js
const form = document.getElementById('insurance-form');

form.addEventListener('submit', async (e) => {
  e.preventDefault();

  const formData = new FormData(form);
  const data = Object.fromEntries(formData);

  // Wyślij do backendu lub Google Sheets
  const response = await fetch('/api/submit', {
    method: 'POST',
    headers: { 'Content-Type': 'application/json' },
    body: JSON.stringify(data)
  });

  if (response.ok) {
    showThankYouMessage();
  }
});
```

4.3 Backend (opcjonalnie - Next.js API)

```
// pages/api/submit.js
export default async function handler(req, res) {
  if (req.method === 'POST') {
    const data = req.body;

    // Zapisz do bazy danych lub wyślij email
    await saveToDatabase(data);

    res.status(200).json({ success: true });
  }
}
```

Krok 5: Hosting

Opcja A: Vercel (rekomendowane dla Next.js)

```
# 1. Zainstaluj Vercel CLI
npm install -g vercel

# 2. Zaloguj się
vercel login

# 3. Deploy
vercel --prod

# 4. Podłącz domenę w panelu Vercel
```

Opcja B: Netlify (dla statycznych stron)

```
# 1. Zainstaluj Netlify CLI
npm install -g netlify-cli

# 2. Deploy
netlify deploy --prod
```

Opcja C: GitHub Pages (darmowe, proste)

```
# 1. Utwórz repo na GitHub  
# 2. Wrzuć pliki  
git push origin main  
  
# 3. Settings > Pages > Source: main branch  
# Strona dostępna: username.github.io/repo-name
```

Opcja D: VPS (pełna kontrola)

- OVH VPS - od 20 zł/mies
- DigitalOcean - od \$5/mies
- Hetzner - od 4€/mies

```
# Na serwerze:  
sudo apt update  
sudo apt install nginx  
sudo certbot --nginx -d twoja-domena.pl
```

Krok 6: SSL (HTTPS)

Automatyczne (Vercel/Netlify)

- Certyfikat SSL jest automatycznie włączony

Ręczne (VPS)

```
# Let's Encrypt (darmowy SSL)  
sudo apt install certbot python3-certbot-nginx  
sudo certbot --nginx -d twoja-domena.pl -d www.twoja-domena.pl
```

Krok 7: Konfiguracja DNS

W panelu rejestratora domeny:

Typ	Nazwa	Wartość	TTL
A	@	76.76.21.21	3600 (IP Vercel)
CNAME	www	cname.vercel-dns.com	3600

Część 2: Pozycjonowanie SEO

Poziom 1: Podstawy techniczne (MUST HAVE)

1.1 Plik robots.txt

```
User-agent: *\nAllow: /\nDisallow: /api/\nDisallow: /admin/\n\nSitemap: https://twoja-strona.pl/sitemap.xml
```

1.2 Sitemap XML

```

<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    <loc>https://strona.pl/</loc>
    <lastmod>2024-01-15</lastmod>
    <changefreq>weekly</changefreq>
    <priority>1.0</priority>
  </url>
  <url>
    <loc>https://strona.pl/ubezpieczenia-na-zycie</loc>
    <lastmod>2024-01-15</lastmod>
    <changefreq>weekly</changefreq>
    <priority>0.9</priority>
  </url>
</urlset>

```

1.3 Google Search Console

1. Wejdź na <https://search.google.com/search-console>
2. Dodaj domenę
3. Zweryfikuj przez DNS (dodaj rekord TXT)
4. Wyślij sitemap.xml
5. Monitoruj indeksację

1.4 Google Analytics 4

```

<!-- Dodaj do <head> -->
<script async src="https://www.googletagmanager.com/gtag/js?id=G-XXXXXXX"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());
  gtag('config', 'G-XXXXXXX');
</script>

```

Poziom 2: Optymalizacja On-Page

2.1 Struktura nagłówków

```

<h1>Ubezpieczenia na życie - porównaj najlepsze oferty 2024</h1>
<h2>Czym jest ubezpieczenie na życie?</h2>
  <h3>Rodzaje ubezpieczeń</h3>
  <h3>Koszty ubezpieczenia</h3>
<h2>Jak wybrać najlepszą polisę?</h2>
<h2>Wypełnij ankietę - bezpłatna wycena</h2>

```

2.2 Optymalizacja obrazów

```



```

2.3 Internal linking

```

<!-- Na każdej podstronie linkuj do głównej -->
<a href="/ubezpieczenia-na-zycie">ubezpieczenie na życie</a>

<!-- W artykułach blogowych -->
<p>Dowiedz się więcej o <a href="/ubezpieczenia-na-zycie">ubezpieczeniach na życie</a>.</p>

```

2.4 Schema.org (Structured Data)

FAQ Schema (daje rich snippets!)

```

<script type="application/ld+json">
{
    "@context": "https://schema.org",
    "@type": "FAQPage",
    "mainEntity": [
        {
            "@type": "Question",
            "name": "Ile kosztuje ubezpieczenie na życie?",
            "acceptedAnswer": {
                "@type": "Answer",
                "text": "Ubezpieczenie na życie kosztuje średnio od 50 do 200 zł miesięcznie, w zależności od wieku, sumy ubezpieczenia i polityki ubezpieczenia."
            }
        },
        {
            "@type": "Question",
            "name": "Czy ubezpieczenie na życie jest obowiązkowe?",
            "acceptedAnswer": {
                "@type": "Answer",
                "text": "Nie, ubezpieczenie na życie jest dobrowolne, ale zalecane szczególnie dla osób z kredytem hipotecznym lub rynkiem nieruchomości."
            }
        }
    ]
}
</script>

```

LocalBusiness Schema (jeśli masz firmę)

```

<script type="application/ld+json">
{
    "@context": "https://schema.org",
    "@type": "InsuranceAgency",
    "name": "Nazwa Firmy",
    "address": {
        "@type": "PostalAddress",
        "streetAddress": "ul. Przykładowa 1",
        "addressLocality": "Warszawa",
        "postalCode": "00-001"
    },
    "telephone": "+48123456789"
}
</script>

```

Poziom 3: Content Marketing

3.1 Strategia long-tail keywords

Zamiast konkurować o "ubezpieczenia na życie", celuj w:

Fraza long-tail	Trudność	Potencjał
najtańsze ubezpieczenie na życie 2024	Średnia	Wysoki

Ubezpieczenie na życie dla 30-latka	Niechęć	Potencjał
ubezpieczenie na życie online bez badań	Niska	Wysoki
ile kosztuje ubezpieczenie na życie	Średnia	Wysoki
ubezpieczenie na życie a kredyt hipoteczny	Niska	Średni
ubezpieczenie na życie porównanie firm	Średnia	Wysoki

3.2 Plan contentu (blog)

Miesiąc 1:

- "Ile kosztuje ubezpieczenie na życie w 2024?" (2000+ słów)
- "Top 10 ubezpieczycieli w Polsce - ranking" (2500+ słów)
- "Ubezpieczenie na życie a kredyt hipoteczny" (1500+ słów)

Miesiąc 2:

- "Jak wybrać ubezpieczenie na życie? Poradnik" (3000+ słów)
- "Ubezpieczenie na życie dla rodziny - co warto wiedzieć" (2000+ słów)
- "Ubezpieczenie grupowe vs indywidualne" (1500+ słów)

Miesiąc 3:

- "Kiedy ubezpieczenie na życie się nie opłaca?" (1500+ słów)
- "Ubezpieczenie na życie z UFK - czy warto?" (2000+ słów)
- Aktualizacja starszych artykułów

3.3 AI do generowania treści

```

# content_generator.py
from openai import OpenAI

def generate_article(keyword):
    client = OpenAI()

    prompt = f"""
Napisz profesjonalny artykuł SEO na temat: {keyword}

Wymagania:
- Minimum 2000 słów
- Użyj nagłówków H2 i H3
- Dodaj FAQ na końcu
- Naturalnie użyj słów kluczowych
- Pisz prostym, zrozumiałym językiem
"""

    response = client.chat.completions.create(
        model="gpt-4",
        messages=[{"role": "user", "content": prompt}]
    )

    return response.choices[0].message.content

# Generuj artykuły dla long-tail
keywords = [
    "najtańsze ubezpieczenie na życie 2024",
    "ubezpieczenie na życie dla młodych",
    "jak działa ubezpieczenie na życie"
]

for kw in keywords:
    article = generate_article(kw)
    save_to_file(f"blog/{slugify(kw)}.md", article)

```

Poziom 4: Link Building

4.1 Web 2.0 Backlinks (darmowe)

Utwórz profile i posty z linkami na:

- Medium.com
- LinkedIn (artykuły)
- Quora (odpowiedzi)
- Reddit (r/polska, r/finanse)
- Wykop.pl
- WordPress.com
- Blogger.com
- Tumblr.com

```

# web2_backlinks.py
platforms = {
    "medium": "https://medium.com",
    "linkedin": "https://linkedin.com",
    "wordpress": "https://wordpress.com"
}

def create_backlink_post(platform, title, content, target_url):
    """
    Utwórz post z linkiem do strony głównej
    """
    # Dodaj naturalny link w treści
    content_with_link = f"{content}\n\nWięcej informacji: {target_url}"

    # Publikuj na platformie (wymaga API każdej platformy)
    publish(platform, title, content_with_link)

```

4.2 Guest Posting

1. Znajdź blogi o tematyce finansowej/ubezpieczeniowej
2. Zaproponuj artykuł gościnny
3. W artykule umieść link do swojej strony

4.3 Katalogi firm (NAP consistency)

- Google My Business (najważniejsze!)
- Panorama Firm
- PKT.pl
- Yelp
- Zumi.pl

4.4 Expired Domains (zaawansowane)

```

# expired_domain_finder.py
import whois
from datetime import datetime

def check_domain_expiry(domain):
    try:
        w = whois.whois(domain)
        if w.expiration_date:
            if isinstance(w.expiration_date, list):
                exp = w.expiration_date[0]
            else:
                exp = w.expiration_date

            if exp < datetime.now():
                return True # Domena wygasła
    except:
        return True # Prawdopodobnie wolna
    return False

# Szukaj wygasłych domen związanych z ubezpieczeniami
# które mają istniejące backlinki (sprawdź w Ahrefs/Majestic)

```

Poziom 5: Automatyzacja i Monitoring

5.1 Bot do monitorowania pozycji

```

# rank_tracker.py
from serpapi import GoogleSearch
import csv
from datetime import datetime

def track_rankings(keywords, domain):
    results = []

    for keyword in keywords:
        search = GoogleSearch({
            "q": keyword,
            "location": "Poland",
            "google_domain": "google.pl",
            "gl": "pl",
            "hl": "pl",
            "num": 100
        })

        data = search.get_dict()
        position = None

        for i, result in enumerate(data.get("organic_results", [])):
            if domain in result.get("link", ""):
                position = i + 1
                break

        results.append({
            "date": datetime.now().isoformat(),
            "keyword": keyword,
            "position": position or "100+"
        })

    # Zapisz do CSV
    with open('rankings.csv', 'a') as f:
        writer = csv.DictWriter(f, fieldnames=["date", "keyword", "position"])
        writer.writerows(results)

    return results

# Uruchamiaj codziennie (cron)
keywords = [
    "ubezpieczenia na życie",
    "najtańsze ubezpieczenie na życie",
    "porównanie ubezpieczeń na życie"
]
track_rankings(keywords, "twoja-strona.pl")

```

5.2 Bot do sprawdzania indeksacji

```

# index_checker.py
import requests

def check_google_index(urls):
    indexed = []
    not_indexed = []

    for url in urls:
        # Użyj Google Search API lub scraping
        query = f"site:{url}"
        # sprawdź czy zwraca wyniki

        if is_indexed:
            indexed.append(url)
        else:
            not_indexed.append(url)

    return indexed, not_indexed

```

5.3 Automatyczne social media

```

# social_automation.py
import schedule
import time

def post_to_social(content, link):
    # Twitter/X
    post_to_twitter(content, link)

    # LinkedIn
    post_to_linkedin(content, link)

    # Facebook
    post_to_facebook(content, link)

# Publikuj nowy artykuł automatycznie
schedule.every().day.at("10:00").do(post_daily_content)

while True:
    schedule.run_pending()
    time.sleep(60)

```

Poziom 6: Zaawansowane techniki (Grey/Black Hat)

| **⚠️ OSTRZEŻENIE:** Poniższe techniki są ryzykowne i mogą skutkować karą od Google!

6.1 CTR Manipulation (WYSOKI RYZYKO)

```

# UWAGA: To jest nielegalne i może skutkować banem!
# Pokazuję tylko w celach edukacyjnych

# Symulacja kliknięć z Google używając:
# - Proxy rotation
# - Różne user agents
# - Realistyczne zachowanie użytkownika
# NIE ZALECAM STOSOWANIA

```

6.2 PBN - Private Blog Network (RYZYKOWNE)

Koncept:

1. Kup 10-20 wygasłych domen z backlinkami
2. Postaw na każdej WordPress
3. Dodaj unikalne treści
4. Linkuj do głównej strony

Ryzyko: Google wykrywa i banuje całą sieć

Koszt: 500-2000 zł na start

6.3 Parasite SEO (średnie ryzyko)

Wykorzystaj autorytet dużych platform:

- Medium.com/twoj-artykul (link do strony)
- LinkedIn Articles
- Quora Spaces
- Reddit posts

Te platformy mają wysoki DR i mogą rankować szybciej

Harmonogram i Budżet

Timeline (6 miesięcy)

Tydzień 1-2: Setup

- Kup domenę (~50 zł)
- Stwórz stronę (Next.js/HTML)
- Hosting (Vercel - darmowe)
- SSL (automatyczny)
- Google Search Console
- Google Analytics

Tydzień 3-4: Content bazowy

- Strona główna zoptymalizowana
- 3 artykuły blogowe (long-tail)
- FAQ z Schema.org
- Sitemap i robots.txt

Miesiąc 2-3: Link building

- 10 profili Web 2.0
- Google My Business
- 5 katalogów firm
- 2 guest posty

Miesiąc 4-6: Skalowanie

- 2-3 artykuły tygodniowo
- Monitoring pozycji
- Optymalizacja na podstawie danych
- Social media automation

Budżet

Opcja minimalna (prawie darmowa)

Pozycja	Koszt
Domena .pl	50 zł/rok

Pozycja (Vercel)	Koszt
SSL	0 zł
Narzędzia SEO	0 zł (Google Search Console)
RAZEM	~50 zł/rok

Opcja rekommendowana

Pozycja	Koszt
Domena .pl	50 zł/rok
Hosting VPS	240 zł/rok
Ahrefs/Semrush (monitoring)	0-400 zł/mies
OpenAI API (content)	50-100 zł/mies
RAZEM	~100-500 zł/mies

Oczekiwane rezultaty

Czasookres	Long-tail keywords	Główna fraza
1 miesiąc	Top 50-100	Top 100+
3 miesiące	Top 10-30	Top 50-100
6 miesięcy	Top 5-10	Top 30-50
12 miesięcy	Top 3-5	Top 10-20

Uwaga: Dla frazy "ubezpieczenia na życie" top 10 jest bardzo trudne bez dużego budżetu. Realistycznie można osiągnąć top 10 dla fraz long-tail.

Checklist końcowy

Techniczne SEO

- HTTPS włączone
- Mobile-friendly (responsive)
- Szybkość ładowania < 3s
- Core Web Vitals w normie
- Sitemap.xml wysłany do Google
- robots.txt skonfigurowany
- Schema.org zaimplementowane
- Canonical URLs ustawione

On-Page SEO

- Title tags zoptymalizowane (50-60 znaków)
- Meta descriptions (150-160 znaków)
- Nagłówki H1-H3 ze słowami kluczowymi
- Obrazy z atrybutami alt
- Internal linking
- URL-e przyjazne SEO

Off-Page SEO

- Google My Business
- Profile Web 2.0
- Katalogi firm
- Social media
- Guest posty

Monitoring

- Google Search Console
 - Google Analytics 4
 - Tracking pozycji (weekly)
 - Monitoring backlinków
-

Dokument wygenerowany: 2024 Autor: AI Assistant