QR Code Generator Website — Full Project

This canvas contains a production-ready scaffold for a **QR Code Generator Website** built with **HTML, CSS, and JavaScript** (vanilla). It includes: project structure, complete code files, 10 pre-written blog posts (each ~550–650 words) with suggested royalty-free image links, Google AdSense placeholder snippets, and deployment instructions for GitHub Pages / Vercel.

> **How to use:** open the code blocks below, copy files into the corresponding folders in your repository (`/src`, `/public`, `/blog`), and follow the README instructions at the top.

Repository structure

```
qr-code-generator-website/
├── README.md
├── public/
│   └── assets/ (put logos & blog images here)
├── src/
│   ├── index.html
│   ├── styles.css
│   └── script.js
└── blog/
    ├── 01-what-is-qr-code.md
    ├── 02-history-and-evolution.md
    ├── 03-qr-codes-in-marketing.md
    ├── 04-designing-effective-qr.md
    ├── 05-security-and-privacy.md
    ├── 06-qr-codes-for-events.md
    ├── 07-qr-code-tracking-analytics.md
    ├── 08-integrating-with-ecommerce.md
    ├── 09-future-of-qr-and-nfc.md
    └── 10-case-studies-success-stories.md
```

README.md

```
# QR Code Generator Website

A clean, minimalist QR Code Generator website (white background) built with
plain HTML, CSS, and JavaScript. Features include color selection, logo-in-
center, download as PNG/SVG, customization for size/margin/error-correction, and
a blog section with 10 pre-written articles. Google AdSense placeholders are
```

included in the blog templates.

Features
- White, minimal design
- Default QR color: Magenta (#FF00FF)
- Color picker for foreground/background
- Upload logo to embed in center
- Download as PNG or SVG
- Size, margin, error-correction control
- Blog with 10 articles (pre-written)
- Google AdSense placeholders for monetization

Tech
- HTML / CSS / JavaScript (vanilla)
- `qr-code-styling` library (via CDN)
- Tailwind-like minimal CSS included

Folder layout
See repository structure in the project root.

Setup & Run
1. Clone or copy files to a new repo.
2. Serve `src/index.html` using a static server or open in browser.

Optionally deploy to GitHub Pages or Vercel. For GitHub Pages: push `main` branch and set Pages source to `/root` or `/docs` depending on where you put files.

Notes
- Replace the placeholder AdSense code with your own AdSense `ca-pub-...` after account approval.
- Store royalty-free blog images in `/public/assets` and update blog frontmatter image path if desired.

---

src/index.html

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width,initial-scale=1">
  <title>QR Factory — Minimal QR Code Generator</title>
  <meta name="description" content="Generate custom QR codes (logo, color,
download PNG/SVG). Clean, minimal design.">
```

```html
<!-- Simple CSS reset + styles -->
<link rel="stylesheet" href="styles.css">

<!-- QR Code Styling library (browser build) -->
<script src="https://cdn.jsdelivr.net/npm/qr-code-styling@1.5.0/lib/qr-code-
styling.js"></script>
</head>
<body>
  <nav class="nav">
    <div class="nav-inner">
      <a class="logo" href="#">QR Factory</a>
      <ul class="nav-menu">
        <li><a href="#generator">Generator</a></li>
        <li><a href="/blog/index.html">Blog</a></li>
      </ul>
    </div>
  </nav>

  <main class="container">
    <section id="generator" class="card">
      <h1>QR Code Generator</h1>
      <div class="grid">
        <div class="controls">
          <label>Text or URL
            <textarea id="text-input" placeholder="Enter URL or text to encode"
rows="3"></textarea>
          </label>

          <div class="row">
            <label>Size
              <input type="range" id="size" min="128" max="1024" value="300">
            </label>
            <span id="size-val">300px</span>
          </div>

          <div class="row">
            <label>Margin
              <input type="range" id="margin" min="0" max="40" value="10">
            </label>
            <span id="margin-val">10</span>
          </div>

          <label>Error correction
            <select id="error-correction">
              <option value="L">Low (L)</option>
              <option value="M" selected>Medium (M)</option>
              <option value="Q">Quartile (Q)</option>
```

```html
          <option value="H">High (H)</option>
        </select>
      </label>

      <label>Foreground color
        <input type="color" id="fg-color" value="#FF00FF">
      </label>

      <label>Background color
        <input type="color" id="bg-color" value="#ffffff">
      </label>

      <label>Upload logo (center)
        <input type="file" id="logo-input" accept="image/*">
      </label>

      <div class="actions">
        <button id="generate">Generate</button>
        <button id="download-png">Download PNG</button>
        <button id="download-svg">Download SVG</button>
      </div>
    </div>

    <div class="preview">
      <div id="qr-wrap"></div>
      <p class="hint">Default color is <strong>Magenta</strong> (#FF00FF).
Use the color pickers to change foreground/background.</p>
    </div>
  </div>
</section>

<section class="card blog-intro">
  <h2>From the Blog</h2>
  <p>Explore articles about QR codes, design tips, analytics, and real-world
uses.</p>
  <a class="btn" href="/blog/index.html">Open Blog</a>
</section>
</main>

<footer class="footer">
  <p>© QR Factory — Clean QR codes, made simple.</p>
</footer>

<!-- App script -->
<script src="script.js"></script>
</body>
</html>
```

src/styles.css

```css
:root{--bg:#ffffff;--text:#111827;--muted:#6b7280;--accent:#ff00ff}
*{box-sizing:border-box}
html,body{height:100%}
body{margin:0;font-family:Inter, ui-sans-serif, system-ui, -apple-system,
'Segoe UI', Roboto, 'Helvetica Neue', Arial;color:var(--text);background:var(--
bg);}
.nav{border-bottom:1px solid #f3f4f6;background:transparent}
.nav-inner{max-width:1100px;margin:0 auto;padding:12px 20px;display:flex;align-
items:center;justify-content:space-between}
.logo{font-weight:700;color:var(--text);text-decoration:none}
.nav-menu{list-style:none;margin:0;padding:0;display:flex;gap:16px}
.nav-menu a{text-decoration:none;color:var(--muted)}
.container{max-width:1100px;margin:28px auto;padding:0 20px}
.card{background:#fff;border:1px solid #f3f4f6;padding:20px;border-radius:
12px;margin-bottom:20px}
.grid{display:grid;grid-template-columns:1fr 360px;gap:24px}
.controls label, .controls .row{display:block;margin-bottom:12px;font-size:
14px;color:var(--muted)}
.controls textarea{width:100%;padding:10px;border:1px solid #e6e6e6;border-
radius:8px}
.controls input[type=range]{width:100%}
.controls input[type=color]{width:48px;height:36px;padding:
0;border:none;background:none}
.actions{display:flex;gap:10px;margin-top:8px}
button{background:var(--accent);color:#fff;border:none;padding:10px 12px;border-
radius:8px;cursor:pointer}
button.secondary{background:#f3f4f6;color:var(--text)}
.preview{display:flex;flex-direction:column;align-items:center;gap:12px}
#qr-wrap{background:transparent;padding:12px;border-radius:8px}
.hint{font-size:13px;color:var(--muted)}
.footer{text-align:center;padding:30px 0;color:var(--muted)}
.blog-intro .btn{display:inline-block;margin-top:12px;padding:8px
12px;background:#111827;color:#fff;border-radius:8px;text-decoration:none}
@media (max-width:900px){.grid{grid-template-columns:1fr}.nav-inner{padding:
12px}.controls{order:2}}
```

src/script.js

```javascript
// QR Generator App (vanilla)
const qrWrap = document.getElementById('qr-wrap');
```

```javascript
const textInput = document.getElementById('text-input');
const sizeInput = document.getElementById('size');
const sizeVal = document.getElementById('size-val');
const marginInput = document.getElementById('margin');
const marginVal = document.getElementById('margin-val');
const fgColor = document.getElementById('fg-color');
const bgColor = document.getElementById('bg-color');
const logoInput = document.getElementById('logo-input');
const generateBtn = document.getElementById('generate');
const downloadPng = document.getElementById('download-png');
const downloadSvg = document.getElementById('download-svg');
const errorSel = document.getElementById('error-correction');

let qrCode = null;
let currentLogo = null;

function createQR() {
  const data = (textInput.value || '').trim() || 'https://example.com';
  const size = parseInt(sizeInput.value, 10) || 300;
  const margin = parseInt(marginInput.value, 10) || 10;
  const fg = fgColor.value || '#FF00FF';
  const bg = bgColor.value || '#FFFFFF';
  const ecLevel = errorSel.value || 'M';

  // clear container
  qrWrap.innerHTML = '';

  // create QRCodeStyling instance
  qrCode = new QRCodeStyling({
    width: size,
    height: size,
    margin: margin,
    data,
    image: currentLogo,
    dotsOptions: {color: fg, type: 'rounded'},
    backgroundOptions: {color: bg},
    qrOptions: {errorCorrectionLevel: ecLevel}
  });

  qrCode.append(qrWrap);
}

// initial default
sizeVal.textContent = sizeInput.value + 'px';
marginVal.textContent = marginInput.value;

sizeInput.addEventListener('input', () => { sizeVal.textContent =
sizeInput.value + 'px'; });
```

```javascript
  marginInput.addEventListener('input', () => { marginVal.textContent =
marginInput.value; });

  logoInput.addEventListener('change', async (e) => {
    const f = e.target.files[0];
    if (!f) return;
    const reader = new FileReader();
    reader.onload = () => {
      currentLogo = reader.result;
      createQR();
    };
    reader.readAsDataURL(f);
  });

  generateBtn.addEventListener('click', (e)=>{ e.preventDefault(); createQR(); });

  downloadPng.addEventListener('click', async ()=>{
    if (!qrCode) createQR();
    const blob = await qrCode.getRawData('png');
    const url = URL.createObjectURL(blob);
    const a = document.createElement('a');
    a.href = url; a.download = 'qr.png';
    document.body.appendChild(a); a.click(); a.remove();
    URL.revokeObjectURL(url);
  });

  downloadSvg.addEventListener('click', async ()=>{
    if (!qrCode) createQR();
    const svg = await qrCode.getRawData('svg');
    const blob = new Blob([svg], {type: 'image/svg+xml'});
    const url = URL.createObjectURL(blob);
    const a = document.createElement('a');
    a.href = url; a.download = 'qr.svg';
    document.body.appendChild(a); a.click(); a.remove();
    URL.revokeObjectURL(url);
  });

  // create on load with defaults
  window.addEventListener('load', createQR);
```

Google AdSense placeholder

Add this snippet into blog layout pages where indicated. Replace `ca-pub-XXXXXXXXXXXX` with your own
AdSense publisher ID after approval.

```
<!-- Google AdSense placeholder -->
<script async src="https://pagead2.googlesyndication.com/pagead/js/
adsbygoogle.js"></script>
<!-- blog-sidebar-1 -->
<ins class="adsbygoogle"
     style="display:block"
     data-ad-client="ca-pub-XXXXXXXXXXXX"
     data-ad-slot="0000000000"
     data-ad-format="auto"
     data-full-width-responsive="true"></ins>
<script>
    (adsbygoogle = window.adsbygoogle || []).push({});
</script>
```

Important: Keep this placeholder code until you receive your actual AdSense ID. DO NOT use a real AdSense ID until your site meets Google policies and you have an account.

Blog index (simple)

Create `/blog/index.html` that lists posts (simple static page linking to markdown or rendered html). For brevity, the canvas includes the markdown posts; you may convert them to HTML or serve via a static site generator.

Blog posts (10) — markdown files

Note: Each article includes a suggested royalty-free image URL (Unsplash or Pexels). Download the image into `public/assets/` and update paths if you prefer.

01-what-is-qr-code.md

```
---
title: What is a QR Code? — A Beginner's Guide
image: ../public/assets/qr-basics.jpg
---

Quick summary: QR codes (Quick Response codes) are two-dimensional barcodes
invented in Japan for tracking automotive parts and now widely used to encode
URLs, contact details, Wi-Fi credentials, and more. In this article we'll
explore what a QR code is, how it encodes information, and practical everyday
uses.

QR codes encode data using black-and-white square modules arranged in a square
```

grid. Scanners — including smartphone cameras — read the pattern and decode it into the original data. Unlike traditional barcodes, which are one-dimensional and store limited data, QR codes can store hundreds of characters and can use levels of error correction to remain readable even when partially damaged.

Why have QR codes become so popular? Their simplicity is one reason: any smartphone with a camera can scan a QR code without extra hardware. Also, QR codes can conveniently encode a URL so users can open a webpage directly, rather than typing a long address. For retailers, event organizers, and marketers, QR codes reduce friction: they move users from physical media to digital interaction with a single scan.

Several practical applications make QR codes useful: contactless menus at restaurants, quick Wi-Fi sharing without typing passwords, event tickets, product packaging linking to instructions, and even business cards that automatically add contact details to a phone. Each of these uses benefits from the speed and convenience of scanning.

It's also important to use QR codes thoughtfully. Make sure the destination URL uses HTTPS, provide clear context nearby (so users know what they'll open), and test the code across different devices and lighting conditions. And when embedding a logo, be mindful of error correction: higher error correction allows for small logos without breaking scannability, but might increase the visual density of the code.

Suggested image: https://unsplash.com/photos/OQMZwNd3ThU (download into `public/assets/qr-basics.jpg`)

02-history-and-evolution.md

```
---
title: The History and Evolution of QR Codes
image: ../public/assets/qr-history.jpg
---
```

The QR code was invented in 1994 by Denso Wave, a subsidiary of Toyota, in Japan. Engineers designed it to track vehicles and parts through the manufacturing process more efficiently than traditional barcodes. The quick-scanning capability and higher data capacity were crucial improvements for automotive logistics.

From manufacturing to mainstream: QR codes gradually moved beyond industry. As smartphone cameras and processing power improved, scanning QR codes became easy for consumers, and their applications broadened. Marketers recognized the value of bridging offline and online experiences: print ads, product packaging, and posters could now point directly at web resources.

The 2010s and smartphone ubiquity turned QR into a near-universal tool. Many countries started using QR codes for public transport payments, digital menus, and contactless services. During the COVID-19 pandemic, QR usage skyrocketed as businesses reduced touch-based interactions. Digital menus, contact tracing, registration forms, and vaccine certificates often used QR for quick interaction.

Standards and improvements have followed. Error correction improvements, visual QR code customization, and adoption of dynamic QR codes (which point to a short redirect URL that can be edited) have made QR codes more versatile. Dynamic QR codes are especially useful for analytics — they allow owners to update destination links without reprinting the code and track scans over time.

Suggested image: https://unsplash.com/photos/mT8L3l6F6zg (save as `public/assets/qr-history.jpg`)

03-qr-codes-in-marketing.md

```
---
title: How QR Codes are Revolutionizing Marketing
image: ../public/assets/qr-marketing.jpg
---
```

QR codes provide a direct channel from physical assets to digital experiences, and that's why marketers love them. Instead of asking customers to type a URL or search a brand, a scan delivers the exact landing page—fast. This reduces friction and increases conversion potential.

There are many creative uses: QR codes embedded on product packaging can open how-to videos, reveal promotional discounts, or present extended product specifications. In retail stores, QR codes near displays can show reviews or allow customers to add items to their online cart. Outdoor campaigns on billboards or transit shelter ads can link to exclusive content or event registration pages.

Best practices include: position the code where it's easily scannable, provide a short instruction (e.g., "Scan to get 10% off"), ensure the landing page is mobile-optimized, and consider using a dynamic QR code to change the landing URL when campaigns evolve. Tracking and analytics are essential—use a URL shortener or redirect service with tracking to measure performance and iterate on placement and messaging.

Suggested image: https://unsplash.com/photos/3Mhgvrk4tjM (save as `public/assets/qr-marketing.jpg`)

04-designing-effective-qr.md

```
---
title: Designing Effective and Beautiful QR Codes
image: ../public/assets/qr-design.jpg
---

Design has become a major consideration in QR code adoption. A purely black-and-
white matrix works fine, but branded, visually consistent QR codes improve
engagement. Use brand colors thoughtfully: ensure sufficient contrast between
foreground and background for reliable scanning. A common recommendation is to
use dark foreground and light background.

When adding a logo, place it at the center and ensure sufficient error
correction (Q or H) so the code remains scannable. Avoid extreme artistic
distortions that remove alignment patterns. Keep whitespace (quiet zone) around
the code so scanners can detect edges. Also, pick a shape style (dots, rounded
squares) that matches your brand language.

Test codes under real-world conditions: different lighting, camera qualities,
and physical sizes. A QR printed too small or with noisy backgrounds may fail to
scan. For print, maintain at least 1 cm minimum size depending on distance and
scanning device.

Suggested image: https://unsplash.com/photos/8manzosu1SE (save as `public/
assets/qr-design.jpg`)
```

05-security-and-privacy.md

```
---
title: QR Codes and Security — What to Watch Out For
image: ../public/assets/qr-security.jpg
---

QR codes themselves are just a convenient container for data, but they can be
abused. A malicious QR could direct users to phishing sites, download harmful
files, or lead to unwanted actions. Encourage users to preview links when
possible and ensure destination URLs use HTTPS.

For site owners, use safety best practices: sanitize and validate any data
captured from QR-driven forms. If using short links or redirection, clearly
communicate the destination and avoid sending users to unexpected or third-party
downloads. Consider adding an intermediate interstitial page that shows the
destination and explains why they were redirected.

Analytics and privacy: many organizations track scans to measure marketing
```

performance. If you collect personal data, maintain clear privacy notices, use secure storage, and comply with relevant regulations (GDPR, CCPA). Anonymize analytics where possible and avoid storing sensitive information directly in the QR code.

Suggested image: https://unsplash.com/photos/3Kv48NS4WUU (save as `public/assets/qr-security.jpg`)

06-qr-codes-for-events.md

```
---
title: Using QR Codes for Events — Tickets, Check-In & More
image: ../public/assets/qr-events.jpg
---

Events benefit hugely from QR-driven workflows: tickets encoded as QR codes speed up check-in and reduce fraud via unique identifiers. Event organizers can embed attendee details or ticket IDs, and scanners at entry can validate tickets against a backend database.

Beyond tickets, QR codes can be used at exhibitor booths to share contact information, link to speaker bios, or allow attendees to download slide decks and resources. Use dynamic QR codes if you plan to update links after printing materials.

Suggested image: https://unsplash.com/photos/wawEfYdpkag (save as `public/assets/qr-events.jpg`)
```

07-qr-code-tracking-analytics.md

```
---
title: Tracking & Analytics for QR Code Campaigns
image: ../public/assets/qr-analytics.jpg
---

Tracking QR performance is essential when used in campaigns. Use dynamic short URLs or redirect services that log scan times, locations (approximate), device types, and other metrics. Connect QR scan data with your analytics stack to measure conversions and ROI.

When using third-party tracking, ensure data privacy compliance and inform users if any personal data is being collected. Segment campaigns by location, creative asset, or print run to compare performance across channels and iterate quickly.
```

Suggested image: https://unsplash.com/photos/6anudmpILw4 (save as `public/assets/qr-analytics.jpg`)

08-integrating-with-ecommerce.md

```
---
title: Integrating QR Codes with E-commerce
image: ../public/assets/qr-ecommerce.jpg
---
```

Ecommerce sites can use QR codes for shopping continuity—an in-store QR directs shoppers to a product page with reviews and alternative SKUs. Packaging can link to a registration page for warranty, or to loyalty program enrollment. Use QR codes on receipts to request feedback or provide promotional codes.

Suggested image: https://unsplash.com/photos/tZc3vjPCk-Q (save as `public/assets/qr-ecommerce.jpg`)

09-future-of-qr-and-nfc.md

```
---
title: The Future of QR Codes and NFC — Complementary Technologies
image: ../public/assets/qr-future.jpg
---
```

QR codes and NFC each have strengths. NFC offers tap interactions on capable devices but requires hardware support; QR codes remain universal due to camera ubiquity. Expect more hybrid solutions where a printed asset includes both a QR and an NFC tag.

Suggested image: https://unsplash.com/photos/KY0E7VhQeVg (save as `public/assets/qr-future.jpg`)

10-case-studies-success-stories.md

```
---
title: QR Code Case Studies — Real-World Success Stories
image: ../public/assets/qr-cases.jpg
---
```

This article shares a few short case studies of organizations using QR codes successfully: a restaurant switching to contactless menus that boosted online orders, a museum using QR-guided audio tours, and a retailer using QR codes on packaging to increase product registration rates.

```
Suggested image: https://unsplash.com/photos/iq5c6zVZJpM (save as `public/
assets/qr-cases.jpg`)
```

Next steps & deployment

1. Copy files into your repository exactly as organized.
2. Populate `public/assets` with image files (download from the suggested Unsplash URLs or use your own royalty-free images).
3. Optionally convert markdown posts to HTML pages (or use a static site generator like Eleventy/Hugo) and place them under `/blog` as `.html` files. A simple approach is to create `/blog/index.html` with links to each markdown page converted to HTML.
4. Replace AdSense `ca-pub-XXXXXXXXXXXX` with your own when ready.
5. Push to GitHub and deploy to GitHub Pages or Vercel.

End of project scaffold.