Google Merchant Center Compliance Checker for Shopify Stores

Web Crawler Requirements Specification (Domain-Only Access)

Executive Summary

This document specifies requirements for a web-based compliance checker that can identify Google Merchant Center (GMC) compliance issues by crawling only the public-facing Shopify store domain. The tool requires no authentication, API access, or backend integration. All checks are performed through standard web crawling techniques, analyzing publicly accessible HTML, CSS, JavaScript, and resources.

---

1. ACHIEVABLE COMPLIANCE CHECKS

1.1 Technical Infrastructure Checks

SSL Certificate Validation ✅

What we can check:

- Verify HTTPS protocol on all pages
- Check SSL certificate validity and expiration
- Identify mixed content warnings (HTTP resources on HTTPS pages)
- Verify SSL on checkout pages (if accessible without login)

How to implement:

- Check URL protocol (https://)
- Validate SSL certificate chain
- Scan page resources for non-HTTPS URLs
- Test common checkout URLs (/checkout, /cart)

Page Performance Metrics ✅

What we can check:

- Page load time measurements
- Core Web Vitals (LCP, FID/INP, CLS)
- Resource size analysis (images, scripts, stylesheets)
- Mobile vs desktop performance

How to implement:

- Use headless browser with performance timing API
- Measure resource loading times
- Calculate cumulative layout shift
- Test with mobile user agent

Mobile Responsiveness ✅

What we can check:

- Viewport meta tag presence
- Responsive CSS detection
- Mobile-friendly navigation elements
- Touch target sizes
- Horizontal scrolling issues

How to implement:

- Parse viewport meta tags
- Test page rendering at different viewport sizes
- Check CSS media queries
- Measure clickable element dimensions

Crawlability and Indexability ✅

What we can check:

- Robots.txt accessibility and rules
- Meta robots tags
- Sitemap.xml presence and validity
- Canonical URL implementation
- No-index/no-follow directives

How to implement:

- Fetch and parse robots.txt
- Check meta robots tags in HTML head
- Validate sitemap.xml structure

- Verify canonical URLs match expected patterns

1.2 Content and Structure Validation

Structured Data Detection ✅

What we can check:

- Product schema markup (JSON-LD or Microdata)
- Organization schema
- BreadcrumbList schema
- Review/Rating schema
- Schema validation errors

How to implement:

- Parse JSON-LD scripts in HTML
- Extract Microdata attributes
- Validate against schema.org specifications
- Check for required Product properties

Store Information Pages ✅

What we can check:

- Return/Refund policy page existence
- Privacy policy page existence
- Terms of service page existence
- Contact information page existence
- Shipping information page existence

How to implement:

- Check common policy URLs (/policies/, */pages/*)
- Search footer for policy links
- Parse page content for policy keywords
- Verify pages return 200 status codes

Contact Information Visibility ✅

What we can check:

- Email addresses on site
- Phone numbers displayed
- Physical address presence
- Contact form availability
- Social media links

How to implement:

- Regex pattern matching for emails/phones
- Search for common contact page URLs
- Parse footer and header for contact info
- Check for mailto: and tel: links

1.3 Product Page Analysis

Product Information Completeness ✅

What we can check:

- Product title presence and length
- Product description availability
- Price display and format
- Product images (count and quality)
- Availability status indicators

How to implement:

- Parse product page HTML structure
- Extract product data from structured data
- Analyze image URLs and dimensions
- Check for price and availability elements

Image Quality Assessment ✅

What we can check:

- Image resolution and dimensions
- File format (JPEG, PNG, WebP)

- Alt text presence
- Number of product images
- Image loading performance

How to implement:

- Fetch image headers for dimensions
- Check img tag attributes
- Measure image file sizes
- Validate image URLs accessibility

Price Display Consistency ✅

What we can check:

- Price format and currency symbols
- Sale price vs regular price display
- Currency consistency across pages
- Price in structured data matching displayed price

How to implement:

- Extract prices using regex patterns
- Compare structured data prices with displayed prices
- Detect currency symbols and codes
- Check for hidden fees in checkout flow (if accessible)

1.4 Shopify-Specific Detection

Theme Identification ✅

What we can check:

- Shopify theme name and ID
- Theme version
- Custom vs standard theme detection

How to implement:

- Search for "Shopify.theme" in page source

- Extract theme_store_id from JavaScript
- Parse theme assets URLs
- Check for theme-specific markers

Shopify Platform Confirmation ✅

What we can check:

- Shopify platform indicators
- CDN usage (cdn.shopify.com)
- Shopify-specific meta tags
- Payment provider detection

How to implement:

- Check for Shopify JavaScript objects
- Verify CDN URLs patterns
- Parse platform-specific meta tags
- Detect Shopify checkout URLs

---

2. PARTIALLY ACHIEVABLE CHECKS

2.1 Limited Product Data Validation

Product Identifiers (Partial) ⚠️

What we can check:

- GTIN/EAN/UPC format validation if displayed
- Brand name presence
- SKU visibility

Limitations:

- Cannot verify if GTINs are valid/real
- Cannot check MPN if not displayed
- Cannot determine identifier_exists value

2.2 Checkout Process (Limited) ⚠️

What we can check:

- Guest checkout option visibility
- Accepted payment methods displayed
- Shipping calculator presence

Limitations:

- Cannot complete full checkout flow
- Cannot verify actual payment processing
- Limited access without adding items to cart

---

3. NON-ACHIEVABLE CHECKS

3.1 Backend Data Validation ❌

Cannot check without API access:

- Complete product feed data
- Inventory synchronization accuracy
- Tax configuration settings
- Shipping rate calculations
- Multi-location inventory
- Product variant relationships (item_group_id)
- Backend-only product attributes

3.2 Merchant Account Verification ❌

Cannot verify:

- Business registration details
- Merchant account standing
- Previous GMC violations
- Account verification status
- Business type classification

3.3 Dynamic Pricing and Inventory ❌

Cannot accurately assess:

- Real-time inventory levels

- Dynamic pricing rules
- Currency conversion accuracy
- Regional pricing variations
- Bulk pricing rules

3.4 Restricted Content Detection ❌

Limited capability for:

- Comprehensive prohibited product detection
- Counterfeit product identification
- Age-restricted content verification
- Trademark violation detection
- Medical/health claims validation

Note: Basic keyword detection is possible but not comprehensive

3.5 Third-Party Integrations ❌

Cannot detect:

- Installed Shopify apps (backend)
- Third-party service integrations
- Analytics and tracking setup
- Email marketing integrations
- Inventory management systems

---

4. IMPLEMENTATION ARCHITECTURE

4.1 Crawling Strategy

```javascript
```

```
// Core crawling configuration
const crawlerConfig = {
  maxDepth: 3,
  maxPages: 100,
  respectRobotsTxt: true,
  userAgent: 'GMC-Compliance-Checker/1.0',
  timeout: 30000,
  rateLimitDelay: 1000
};
```

4.2 Data Collection Methods

1. **Static HTML Analysis**
   - DOM parsing with cheerio/jsdom
   - Regular expression matching
   - XPath queries for structured data
2. **JavaScript Rendering**
   - Headless browser (Puppeteer/Playwright)
   - Wait for dynamic content loading
   - Execute JavaScript for SPA detection
3. **Resource Analysis**
   - HTTP header inspection
   - Image metadata extraction
   - Performance timing collection

4.3 Compliance Scoring System

javascript

```
const complianceCategories = {
  critical: {
    weight: 10,
    checks: ['ssl', 'pricing', 'policies']
  },
  high: {
    weight: 5,
    checks: ['structured_data', 'mobile', 'images']
  },
  medium: {
    weight: 2,
    checks: ['performance', 'contact_info']
  },
  low: {
    weight: 1,
    checks: ['meta_tags', 'social_links']
  }
};
```

---

5. REPORTING STRUCTURE

5.1 Report Categories

Definitely Fixable Issues

- Missing SSL certificates
- Missing policy pages
- No structured data
- Missing contact information
- Poor mobile responsiveness
- Slow page load times
- Missing/poor quality images

Potentially Present Issues (Requires Manual Verification)

- Product identifier problems
- Pricing inconsistencies
- Checkout process issues
- Content policy violations

Cannot Determine (Requires Backend Access)

- Feed data accuracy
- Inventory sync status
- Tax/shipping configuration
- Account standing issues

5.2 Output Format

```
json

{
  "url": "https://example.myshopify.com",
  "timestamp": "2024-01-15T10:00:00Z",
  "platform": "shopify",
  "theme": "Dawn 2.0",
  "compliance_score": 75,
  "issues": {
    "critical": [...],
    "warnings": [...],
    "suggestions": [...]
  },
  "limitations": [
    "Cannot verify product feed data",
    "Limited checkout process access"
  ]
}
```

---

6. USER EXPECTATIONS MANAGEMENT

6.1 Clear Messaging About Limitations

The tool should clearly communicate:

- "This tool checks only publicly visible compliance issues"
- "Full GMC compliance requires backend verification"
- "Some issues may be false positives requiring manual review"

6.2 Actionable Recommendations

For each detected issue, provide:

- Clear description of the problem
- Why it matters for GMC approval
- Specific steps to fix (when possible)
- Links to relevant Google documentation

6.3 Confidence Levels

Rate each finding with confidence:

- **High Confidence**: Definitely an issue (e.g., no SSL)
- **Medium Confidence**: Likely an issue (e.g., missing structured data)
- **Low Confidence**: Possible issue, needs manual verification

---

7. TECHNICAL REQUIREMENTS

7.1 Performance Requirements

- Complete basic scan in under 60 seconds
- Handle stores with up to 10,000 products
- Process pages concurrently (max 5 simultaneous)
- Cache results for 24 hours

7.2 Error Handling

- Graceful handling of rate limiting
- Timeout recovery mechanisms
- Partial result reporting on failures
- Clear error messages for users

7.3 Compliance and Ethics

- Respect robots.txt directives
- Implement reasonable rate limiting
- Include proper User-Agent identification
- No attempt to bypass security measures

CONCLUSION

This specification provides a realistic framework for GMC compliance checking using only public domain crawling. While it cannot achieve 100% compliance verification, it can identify many common issues that cause GMC disapprovals. The tool should be positioned as a "pre-flight check" that helps merchants identify and fix obvious issues before submitting to Google Merchant Center, with clear communication that additional backend verification may be required for complete compliance.