

# Maison Common Quick Order Plugin v1.3.1

## Technische Dokumentation für Entwickler und IT-Mitarbeiter

---

### Architektur-Übersicht

#### Modularer Aufbau

Das Plugin folgt einer **modularen Architektur** mit klarer Trennung der Verantwortlichkeiten:

```
maison-common-quick-order/
├── maison-common-quick-order.php    # Basis-Plugin (gemeinsame Funktionen)
├── includes/
│   ├── class-mc-order-system.php    # Order-System (bestehende
Funktionalität)
│   └── class-mc-sampling-system.php  # Sampling-System (EAN-Scanner +
Abmusterung)
├── assets/
│   ├── js/
│   │   ├── mc-collections-ajax.js   # Order-System JavaScript
│   │   └── mc-sampling-ajax.js      # Sampling-System JavaScript
│   └── css/
│       ├── mc-collections-style.css  # Order-System Styles
│       └── mc-sampling-style.css     # Sampling-System Styles
└── CHANGELOG.md
```

#### Klassen-Struktur

```
MC_Quick_Order           // Basis-Klasse (gemeinsame Funktionen)
├── MC_Order_System       // Order-System (Collections, Cart Totals)
└── MC_Sampling_System    // Sampling-System (EAN-Scanner,
Abmusterung)
```

**Design-Prinzipien:** - **Single Responsibility:** Jede Klasse hat eine klar definierte Aufgabe - **Dependency Injection:** Systeme sind lose gekoppelt - **WordPress Standards:** Hooks, Filters, und WordPress Coding Standards - **Backward Compatibility:** Bestehende Funktionen bleiben unverändert

---

### System-Requirements

#### Mindestanforderungen

- **WordPress:** 5.0+
- **WooCommerce:** 4.0+

- **PHP:** 7.4+ (empfohlen: 8.0+)
- **MySQL:** 5.6+ oder MariaDB 10.1+

## Plugin-Dependencies

- **WooCommerce Quick Order Plugin:** Für Order-System Tabellen
- **User Switching Plugin:** Optional für Kundenwechsel-Funktionalität

## Browser-Kompatibilität

- **Desktop:** Chrome 70+, Firefox 65+, Safari 12+, Edge 79+
- **Mobile:** iOS Safari 12+, Chrome Mobile 70+
- **JavaScript:** ES6+ Features werden verwendet

## Hardware (Sampling-System)

- **EAN-Scanner:** HID-kompatible Bluetooth-Scanner
- **Unterstützte Formate:** EAN-13, EAN-8, UPC-A
- **Verbindung:** Bluetooth HID oder USB HID

## Installation & Setup

### 1. Plugin-Installation

*# Via WordPress-Backend*

1. Plugin-ZIP hochladen
2. Aktivieren
3. Abhängigkeiten prüfen

*# Via FTP/SFTP*

1. Dateien nach /wp-content/plugins/maison-common-quick-order/ kopieren
2. Berechtigungen setzen (755 für Ordner, 644 für Dateien)
3. Plugin im Backend aktivieren

### 2. Abhängigkeiten installieren

*// Erforderlich für Order-System*

- WooCommerce Quick Order Plugin

*// Optional für erweiterte Funktionen*

- User Switching Plugin (für Kundenwechsel)

### 3. Konfiguration

*Collections für Sampling markieren*

*// Im WordPress-Backend: Produkte → Collections*

1. Collection bearbeiten
2. Checkbox "Abmusterungs-Collection" aktivieren
3. Speichern

## EAN-Felder konfigurieren

Das Plugin sucht automatisch in folgenden Meta-Feldern:

```
$ean_fields = array(  
    '_ean',          // Standard  
    '_ean13',        // EAN-13 spezifisch  
    '_barcode',      // Allgemein  
    '_gtin'          // Global Trade Item Number  
);
```

## EK-Preise einrichten

*// Produktebene: Produkte → Produkt bearbeiten → Preise*

- Feld "EK-Preis (Einkaufspreis)" ausfüllen

*// Varianten-Ebene: Bei variablen Produkten*

- Jede Variante kann eigenen EK-Preis haben

---

## Shortcode-Referenz

### Order-System

`[mc_quick_order]`

Zeigt das Order-System mit Collections-Navigation und Quick Order Tabelle.

*// Basis-Verwendung*

`[mc_quick_order]`

*// Mit spezifischer Collection*

`[mc_quick_order collection_id="123"]`

*// Ohne Collections-Navigation*

`[mc_quick_order show_collections="false"]`

`[mc_cart_totals]`

Zeigt EK/VK-Summen des aktuellen Warenkorbs.

*// Standard-Anzeige*

`[mc_cart_totals]`

*// Nur EK-Summe*

`[mc_cart_totals show_vk="false"]`

*// Nur VK-Summe*

`[mc_cart_totals show_ek="false"]`

*// Custom CSS-Klassen*

`[mc_cart_totals class="custom-totals-style"]`

## Sampling-System

*[mc\_sampling]*

Zeigt das Sampling-System mit EAN-Scanner und Quick Order Integration.

*// Automatische Collection-Erkennung*

*[mc\_sampling]*

*// Spezifische Collection für Vorführungen*

*[mc\_sampling collection\_id="123"]*

*// Nur Scanner ohne Collections-Navigation*

*[mc\_sampling show\_collections="false"]*

*// Nur Scanner ohne Quick Order Tabelle*

*[mc\_sampling show\_table="false"]*

*// Nur Tabelle ohne Scanner*

*[mc\_sampling show\_scanner="false"]*

*// Kombinierte Optionen*

*[mc\_sampling collection\_id="123" show\_collections="false" show\_table="true"]*

---

## AJAX-Endpoints

### Order-System

*mc\_load\_collection*

Lädt Produkte einer spezifischen Collection.

*// Request*

```
{  
  action: 'mc_load_collection',  
  collection_id: 123,  
  nonce: mc_collections_ajax.nonce  
}
```

*// Response*

```
{  
  success: true,  
  data: {  
    html: '<div class="quick-order-table">...</div>',  
    collection_name: 'Frühjahr 2024'  
  }  
}
```

### *mc\_search\_customers*

Sucht Kunden für User Switching Integration.

*// Request*

```
{
  action: 'mc_search_customers',
  search: 'Mustermann',
  nonce: mc_collections_ajax.nonce
}
```

*// Response*

```
{
  success: true,
  data: [
    {
      id: 123,
      display_name: 'Max Mustermann',
      user_email: 'max@example.com'
    }
  ]
}
```

### *mc\_get\_cart\_totals*

Holt aktuelle Warenkorb-Summen.

*// Request*

```
{
  action: 'mc_get_cart_totals',
  nonce: mc_collections_ajax.nonce
}
```

*// Response*

```
{
  success: true,
  data: {
    ek_total: '€ 45,50',
    vk_total: '€ 89,90',
    item_count: 3
  }
}
```

## Sampling-System

### *mc\_sampling\_add\_item*

Fügt Abmusterungs-Item über EAN-Code hinzu.

*// Request*

```
{
  action: 'mc_sampling_add_item',
  ean_code: '1234567890123',
}
```

```
    collection_id: 123,  
    nonce: mc_sampling_ajax.nonce  
}  
  
// Response  
{  
    success: true,  
    data: {  
        message: 'Produkt hinzugefügt',  
        product_name: 'Beispiel-Produkt',  
        artikel_id: 'ART-123'  
    }  
}
```

---

## Datenbank-Schema

### Custom Meta-Fields

#### *Produkt-Meta*

##### *-- EK-Preise*

```
meta_key: '_ek_price'  
meta_value: '45.50'
```

##### *-- EAN-Codes (verschiedene Felder unterstützt)*

```
meta_key: '_ean'  
meta_value: '1234567890123'
```

#### *Collection-Meta*

##### *-- Sampling-Collection Markierung*

```
meta_key: 'is_sampling_collection'  
meta_value: '1'
```

#### *Order-Meta (Sampling-Bestellungen)*

##### *-- Unterschrift erforderlich*

```
meta_key: 'requires_signature'  
meta_value: '1'
```

##### *-- Unterschrift-Timestamp*

```
meta_key: 'signature_timestamp'  
meta_value: '2024-03-15 14:30:00'
```

#### *Cart-Item-Meta (Sampling-Items)*

##### *-- Abmusterungs-Flag*

```
'is_sampling' => true
```

##### *-- Artikel-ID (Parent-SKU)*

```
'artikel_id' => 'ART-123'
```

```
-- Original EAN (bei Scanner-Eingabe)
'scanned_ean' => '1234567890123'

-- Größe-Bezeichnung
'sampling_size' => 'Abmusterung'
```

---

## Integration Points

### WooCommerce Integration

#### Cart-System

```
// Custom Cart Items für Sampling
add_filter('woocommerce_add_cart_item_data',
'mc_add_sampling_cart_item_data');
add_filter('woocommerce_get_item_data',
'mc_display_sampling_cart_item_data');
add_action('woocommerce_checkout_create_order_line_item',
'mc_save_sampling_order_item_meta');
```

#### Pricing-System

```
// EK-Preis Integration
add_action('woocommerce_product_options_pricing', 'mc_add_ek_price_field');
add_action('woocommerce_process_product_meta', 'mc_save_ek_price_field');
add_filter('woocommerce_quick_order_meta_keys',
'mc_add_ek_price_to_meta_keys');
```

#### Checkout-System

```
// Unterschriften-Integration
add_action('woocommerce_checkout_process', 'mc_validate_sampling_signature');
add_action('woocommerce_checkout_update_order_meta',
'mc_save_signature_meta');
```

### Quick Order Plugin Integration

#### Meta-Keys Extension

```
// EK-Preis zu Quick Order Meta-Keys hinzufügen
add_filter('woocommerce_quick_order_meta_keys', function($meta_keys) {
    if (!in_array('_ek_price', $meta_keys)) {
        $meta_keys[] = '_ek_price';
    }
    return $meta_keys;
});
```

#### Table Data Modification

```
// Formatierung der EK-Preise in Quick Order Tabellen
add_filter('woocommerce_quick_order_table_data',
'mc_modify_quick_order_table_data');
```

## User Switching Integration

### Customer Search

*// AJAX-Handler für Kundensuche*

```
add_action('wp_ajax_mc_search_customers', 'mc_ajax_search_customers');
add_action('wp_ajax_nopriv_mc_search_customers', 'mc_ajax_search_customers');
```

---

## Debugging & Troubleshooting

### Debug-Modus aktivieren

*// In wp-config.php*

```
define('WP_DEBUG', true);
define('WP_DEBUG_LOG', true);
define('WP_DEBUG_DISPLAY', false);
```

### Log-Ausgaben

Das Plugin schreibt Debug-Informationen in das WordPress Error-Log:

*// Order-System Logs*

```
error_log('MC Quick Order: ajax_load_collection called');
error_log('MC Quick Order: Collection ID: ' . $collection_id);
```

*// Sampling-System Logs*

```
error_log('MC Sampling: EAN scan attempt: ' . $ean_code);
error_log('MC Sampling: Product found: ' . $product_id);
```

### Häufige Probleme

#### *EAN-Scanner funktioniert nicht*

*// Browser-Konsole prüfen*

```
console.log('Scanner input detected:', eanCode);
```

*// Mögliche Ursachen:*

1. Scanner nicht als HID-Device erkannt
2. JavaScript-Fehler in mc-sampling-ajax.js
3. AJAX-Endpoint nicht erreichbar

#### *Quick Order Tabelle lädt nicht*

*// WordPress Debug-Log prüfen*

1. AJAX-Nonce ungültig
2. Collection-ID nicht gefunden
3. Quick Order Plugin nicht aktiv

#### *EK-Preise werden nicht angezeigt*

*// Mögliche Ursachen:*

1. \_ek\_price Meta-Field nicht gesetzt
2. Quick Order Plugin Cache nicht geleert
3. JavaScript-Formatierung fehlgeschlagen



## Performance-Optimierung

### AJAX-Caching

```
// Collection-Daten cachen
$cache_key = 'mc_collection_' . $collection_id;
$cached_data = get_transient($cache_key);
if ($cached_data === false) {
    // Daten laden und cachen
    set_transient($cache_key, $data, HOUR_IN_SECONDS);
}
```

### Asset-Loading

```
// Conditional Loading nur auf relevanten Seiten
if (has_shortcode($post->post_content, 'mc_quick_order') ||
    has_shortcode($post->post_content, 'mc_sampling')) {
    wp_enqueue_script('mc-collections-ajax');
    wp_enqueue_style('mc-collections-style');
}
```

---

## Security

### Nonce-Validierung

```
// Alle AJAX-Requests verwenden WordPress Nonces
if (!wp_verify_nonce($_POST['nonce'], 'mc_collections_nonce')) {
    wp_die('Security check failed');
}
```

### Input-Sanitization

```
// Alle Eingaben werden sanitized
$collection_id = absint($_POST['collection_id']);
$search_term = sanitize_text_field($_POST['search']);
$ean_code = sanitize_text_field($_POST['ean_code']);
```

### Capability-Checks

```
// Benutzer-Berechtigungen prüfen
if (!current_user_can('edit_shop_orders')) {
    wp_die('Insufficient permissions');
}
```

---

## Wartung & Updates

### Plugin-Updates

```
// Version-Check in Basis-Plugin
if (get_option('mc_quick_order_version') !== MC_QUICK_ORDER_VERSION) {
    // Update-Routine ausführen
}
```

```
mc_run_update_routine();
update_option('mc_quick_order_version', MC_QUICK_ORDER_VERSION);
}
```

## Datenbank-Wartung

```
-- Verwaiste Meta-Daten bereinigen
DELETE pm FROM wp_postmeta pm
LEFT JOIN wp_posts p ON pm.post_id = p.ID
WHERE p.ID IS NULL AND pm.meta_key LIKE '_ek_price';
```

## Cache-Management

```
// Plugin-spezifische Caches leeren
function mc_clear_plugin_cache() {
    global $wpdb;
    $wpdb->query("DELETE FROM {$wpdb->options} WHERE option_name LIKE
'_transient_mc_%'");
}
```

---

## API-Erweiterungen

### Custom Hooks

#### Actions

```
// Sampling-Item hinzugefügt
do_action('mc_sampling_item_added', $cart_item_key, $product_data);

// Collection gewechselt
do_action('mc_collection_switched', $old_collection_id, $new_collection_id);

// Cart Totals berechnet
do_action('mc_cart_totals_calculated', $ek_total, $vk_total);
```

#### Filters

```
// EAN-Felder anpassen
$ean_fields = apply_filters('mc_sampling_ean_fields', $default_ean_fields);

// Sampling-Collections filtern
$collections = apply_filters('mc_sampling_collections', $collections);

// Cart Totals Format anpassen
$formatted_total = apply_filters('mc_cart_totals_format', $total, $type);
```

## REST-API Integration (geplant v1.4.0)

```
// Geplante Endpoints
GET /wp-json/mc/v1/collections
GET /wp-json/mc/v1/sampling/{collection_id}
```

```
POST /wp-json/mc/v1/sampling/add-item
GET /wp-json/mc/v1/cart-totals
```

---

## Deployment

### Staging-Environment

```
# Plugin-Dateien kopieren
rsync -av --exclude='.git' maison-common-quick-order/ staging/wp-
content/plugins/maison-common-quick-order/
```

```
# Datenbank-Migration
wp db export production.sql
wp db import production.sql --url=staging.example.com
```

### Production-Deployment

```
# Backup erstellen
wp db export backup-$(date +%Y%m%d).sql

# Plugin aktualisieren
wp plugin update maison-common-quick-order

# Cache Leeren
wp cache flush
```

### Monitoring

```
// Error-Monitoring
add_action('wp_ajax_mc_', function() {
    if (defined('WP_DEBUG') && WP_DEBUG) {
        error_log('MC Plugin AJAX call: ' . $_POST['action']);
    }
});
```

---

## Roadmap

### Version 1.4.0 (Q2 2024)

- **REST-API:** Vollständige API für externe Integration
- **Verkaufsbuch-Export:** PDF-Generierung aus Abmusterungen
- **Erweiterte Scanner-Unterstützung:** Multi-Device-Support

### Version 1.5.0 (Q3 2024)

- **Offline-Modus:** PWA-Features für iPad-Nutzung
- **Advanced Analytics:** Reporting-Dashboard
- **LSE-Connector:** Direkte Warenwirtschafts-Integration

## Version 2.0.0 (Q4 2024)

- **Headless-Architecture:** React-Frontend
  - **Multi-Tenant:** Support für mehrere Mandanten
  - **AI-Integration:** Intelligente Produktvorschläge
- 

## Support-Kontakte

### Entwicklung

- **Lead Developer:** Michael Deinhardt
- **Repository:** Internal GitLab
- **Documentation:** Confluence

### Deployment

- **Staging:** staging.maisoncommon.com
  - **Production:** shop.maisoncommon.com
  - **Monitoring:** New Relic / WordPress Health Check
- 

*Diese Dokumentation wird kontinuierlich aktualisiert. Letzte Änderung: Version 1.3.1 - Juni 2024*