
Itk25NowChildTheme

Entwicklerdokumentation (UltraForce Theme)

André Fischer

02.04.2026

Inhaltsverzeichnis

1	Einordnung	2
2	Plugin-Klasse	2
3	Theme-Vererbung (theme.json)	2
3.1	views (Auflösungsreihenfolge)	3
3.2	style (SCSS-Kette)	3
3.3	asset	3
4	System-Konfiguration (config.xml)	3
4.1	Feld customCss	4
4.2	Feld excludedPages	4
4.2.1	Optionen (IDs und Bezeichnungen aus der XML-Definition)	4
5	Laufzeitverhalten: Custom CSS	4
6	Betrieb und Abhängigkeiten	5
7	Verzeichnisüberblick (relevant für Entwickler)	5

1 Einordnung

- **Anzeige-Label (Shopware):** „ITK UltraForce Theme (Itk25NowChildTheme)“ (laut `composer.json` → `extra.label`).
- **Shopware-Plugin-Klasse:** `Itk25NowChildTheme\\Itk25NowChildTheme`
- **PSR-4:** `Itk25NowChildTheme\\` → `custom/plugins/Itk25NowChildTheme/src/`
- **PHP-Umfang:** Nur die leere Plugin-Klasse; **kein** `services.xml`, **keine** eigenen Controller/Services. Die gesamte Business-Logik liegt im **Parent-Theme-Plugin** `Itk25nowTheme`.

2 Plugin-Klasse

Datei: `custom/plugins/Itk25NowChildTheme/src/Itk25NowChildTheme.php`

```
1 class Itk25NowChildTheme extends Plugin implements ThemeInterface
2 {
3 }
```

Install/Aktivierung führen **keine** zusätzlichen Schritte aus (keine Custom Fields, keine Migrationen in diesem Plugin).

3 Theme-Vererbung (theme.json)

Pfad: `custom/plugins/Itk25NowChildTheme/src/Resources/theme.json`

Schlüssel	Bedeutung
parentTheme	Itk25NowTheme – technischer Name des Parent-Theme-Plugins (muss installiert und als Theme verfügbar sein).
name	Anzeige: „UltraForce Theme (25now Child)“.

3.1 views (Auflösungsreihenfolge)

Reihenfolge im Array (von Shopware von links nach rechts interpretiert; spätere Einträge können frühere überschreiben):

1. @Storefront
2. @Itk25NowTheme
3. @Itk25NowChildTheme
4. @Plugins

Damit überschreibt das Child die Parent-/Storefront-Templates und liegt **vor** @Plugins, sodass Plugin-Templates noch darüber liegen können.

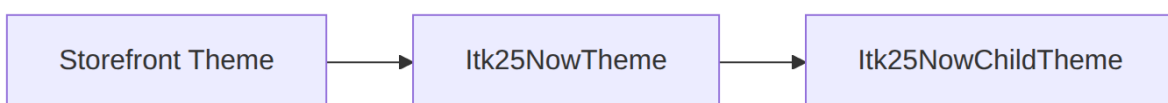
3.2 style (SCSS-Kette)

1. app/storefront/src/scss/overrides.scss (Child)
2. @Storefront
3. @Itk25NowTheme
4. app/storefront/src/scss/base.scss (Child)

Overrides werden zuerst geladen, danach Storefront- und Parent-Styles, zuletzt die Child-base.scss.

3.3 asset

- @Storefront
- @Itk25NowTheme
- app/storefront/src/assets (eigene statische Assets des Child)



4 System-Konfiguration (config.xml)

Pfad: custom/plugins/Itk25NowChildTheme/src/Resources/config/config.xml

Config-Präfix: **Itk25NowChildTheme.config.***

4.1 Feld `customCss`

- **Typ:** `text-editor`
- **Key:** `Itk25NowChildTheme.config.customCss`
- **Zweck:** Freies CSS, das **global** im Storefront ausgegeben werden soll (siehe Einbindung unten).

4.2 Feld `excludedPages`

- **Typ:** `multi-select`
- **Key:** `Itk25NowChildTheme.config.excludedPages`
- **Zweck:** Liste von **Storefront-Routen-IDs**; auf diesen Routen wird das Custom CSS **nicht** injiziert.

4.2.1 Optionen (IDs und Bezeichnungen aus der XML-Definition)

Route-ID (<code>id</code>)	Bezeichnung (de-DE)
<code>frontend.checkout.confirm.page</code>	Checkout Bestätigung
<code>frontend.checkout.offcanvas</code>	Checkout Offcanvas
<code>frontend.checkout.cart.page</code>	Warenkorb
<code>frontend.checkout.address.page</code>	Checkout Adresse
<code>frontend.account.login.page</code>	Login
<code>frontend.account.register.page</code>	Registrierung

5 Laufzeitverhalten: Custom CSS

Die Ausgabe erfolgt **ohne PHP-Controller** direkt in einem Storefront-Layout-Template des Child-Themes:

- **Datei (technischer Einstieg):** `custom/plugins/Itk25NowChildTheme/src/Resources/views/storefront/layout/meta.html.twig`

Logik:

1. SystemConfig lesen: `customCss` und `excludedPages`.

2. Wenn `customCss` nicht leer ist **und** die aktuelle Route **nicht** in der Liste `excludedPages` vorkommt (bzw. die Liste leer ist), wird der konfigurierte CSS-String im Seitenkopf ausgegeben.
3. Andernfalls wird kein zusätzliches CSS aus dieser Konfiguration eingefügt.

Die genaue HTML-Struktur (z. B. `<style>`) steht ausschließlich in der genannten `.twig`-Datei.

Dieselbe Datei erweitert `@Itk25nowTheme/storefront/layout/meta.html.twig` und kann **weitere Blöcke** enthalten (z. B. Tracking-Skripte); diese werden hier nicht dokumentiert.

6 Betrieb und Abhängigkeiten

1. Plugin **Itk25nowTheme** muss installiert und aktiv sein (Parent-Theme-Plugin mit Custom Fields, Services usw. – siehe `itk25nowtheme-entwicklerdokumentation.md`).
2. Plugin **Itk25NowChildTheme** installieren und aktivieren.
3. Im **Theme-Manager** dem Verkaufskanal das Child-Theme „**UltraForce ...**“ zuweisen und Theme kompilieren (`bin/console theme:compile` bzw. Admin).
4. Nach Änderungen an `customCss` ggf. HTTP-/Reverse-Proxy-Caches und Shopware-Cache beachten.

7 Verzeichnisüberblick (relevant für Entwickler)

Pfad	Inhalt
<code>src/Itk25NowChildTheme.php</code>	Plugin-Bootstrap
<code>src/Resources/theme.json</code>	Vererbung, Assets, Scripts
<code>src/Resources/config/config.xml</code>	Plugin-Einstellungen
<code>src/Resources/views/</code>	Storefront-Overrides (inkl. <code>layout/meta.html.twig</code> für Custom CSS)
<code>src/Resources/app/storefront/...</code>	SCSS-Quellen, gebautes JS unter <code>dist/</code> laut <code>theme.json</code>
