

UI-Kit Livewire 3-> 4

#Upgrades/Livewire

#UI-Kit/upgrade

Ausgangslage

- **Aktuell:** Livewire `^3.6.4`
- **Ziel:** Livewire `^4.0`

Livewire-Footprint im Projekt

Bereich	Umfang
Livewire PHP-Dateien (Components/Traits)	26 Dateien
Blade-Templates mit <code>wire:</code> -Direktiven	~30+ Dateien
Test-Dateien	16 Dateien
Custom Synth (<code>StepSynth</code>)	1
Custom Macros (Component + Test)	4

Genutzte Features

- Lifecycle Hooks (`mount`, `boot`, `hydrate`, `dehydrate`, `updated`)
- `#[On]` und `#[Computed]` Attributes
- `@entangle` (Switch, Dropdown, Step)
- `wire:model`, `wire:click`, `wire:submit.prevent`, `wire:loading`, `wire:key`,
`wire:ignore`, `wire:navigate`, `wire:target`
- `$this->dispatch()` Event-System
- Custom Property Synthesizer (`StepSynth`)
- Custom Component Macros und Test Macros (`emitEvents`, `getStepState`, `assertFocused`)

Breaking Changes (Livewire 3 → 4)

Risiko	Bereich	Betrifft uns?
Mittel	<code>wire:model</code> hört nur noch auf Events direkt am Element	Ja – viele <code>wire:model</code> in Input-Components
Mittel	<code>.blur</code> / <code>.change</code> Modifier-Verhalten geändert	Möglich – prüfen in Inputs
Mittel	Custom Synths API-Änderungen	Ja – <code>StepSynth</code> muss geprüft/angepasst werden
Mittel	Component-Tags müssen korrekt geschlossen werden	Prüfen in Blade-Templates
Niedrig	<code>Route::livewire()</code> als neuer Standard	Nicht direkt relevant (UI-Kit, kein App-Routing)
Niedrig	Config/Method-Signature-Änderungen	Prüfen

Phasen

Phase 1: Upgrade & Fixes (1 Tag)

- ✓ `composer.json` anpassen auf `^4.0`
- ✓ `StepSynth` anpassen (Custom Synth = höchstes Risiko)
- ✓ `wire:model`-Verhalten in allen Input-Components prüfen/fixen (~10+ Templates)
- ✓ `@entangle`-Nutzung prüfen (Switch, Dropdown, Step)
- ✓ Event-Dispatching und `#[On]` Attribute verifizieren
- ✓ Lifecycle Hooks (`boot`, `hydrate`, `dehydrate`, `updated`) auf Kompatibilität prüfen

Phase 3: Testing (1 Tag)

- ✓ Alle 16 Livewire-Testdateien durchlaufen lassen
- ✓ Custom Test-Macros (`emitEvents`, `getStepState`) auf Kompatibilität prüfen
- ✓ Manuelles Testing der komplexeren Components (Stepper, Table)

Phase 4: Downstream-Impact (1-2 Tag)

☐ UI Kit Demo projekt updaten

Gesamtschätzung

Szenario	Aufwand
Best Case (wenige Breaking Changes treffen zu)	2–3 Tage
Realistisch	3–4 Tage
Worst Case	5–6 Tage

Empfehlungen

1. **Machbar:** Ja. Livewire 4 ist abwärtskompatibel designed, die meisten Änderungen sind inkrementell.
2. **Grösstes Risiko:** Der custom `StepSynth` und die `wire:model`-Behavior-Änderungen in den vielen Input-Components.