Programátorská dokumentace

Veškerý kód je v PHP nebo v konfiguraci formátu neon. Všude, kde je to bylo časově možné jsem používal příkladné postupy a dependency injection.
Třídy až na pár vyjímek jsou jednoduché a drží se kontraktu deklarovaných rozhraním.

Specifika

HarvestModule\XmlGenerator

Tato třída generuje výsledné XML za použití definice šablony v konfiguraci a zdrojů, které jsou převážně také deklarovány v konfiguraci. Používá dva specifické zdroje: sklizeň (harvest) nad kterou generování provádí a hodnoty formuláre (form), pokud jsou zadané. Zbytek zdrojů je v definovaný v konfiguraci. Iterator je pseoudo zdroj, který se hodí při generování dětí pomocí tabulkového zdroje.
Možné nastavení jednotlivých zdrojů mají idiomatické názvy a jejich vychozí hodnoty jsou definovány při volání setOptions.

Použité moduly a knihovky

V zdroji je sice composer.json ale projekt používá upravené kódy zejména web-resource-manager a webarchive. Zbytek knihoven by měl být identický se stažením přes composer.

TAR Archívy

Kvulí nevhodné konfiguraci serveru a specifickým podmínkám přístupu k velkým datovým souborům sem napsal vlastní implementaci čtenáře TAR archívů. Měla by být kompatibilní s USTAR formátem. Pro rychlejší eliminaci chyb jsem kód zveřejnil na https://github.com/mishak87/archive-tar

WebArchive

Čtenář hlaviček formátů WARC a ARC. Implementoval jsem v PHP i přesto, že jsem měl fungující implementaci pomocí dostupných knihoven v python. Kvůli časové tísní a konfiguraci produkčního serveru jsem se rozhodl implementovat základní fci v PHP. Implementace je podle specifikací WARC i ARC se základní detekcí špatných archívů.