# Seznam funkčností

## FÁZE I.

Zobrazení bude ve dvou perspektivách, BUFFER a STORE, v poznámkách jednotlivých funkčností bude, v jaké perspektivě se nachází. Obecně, Resources v BUFFER perspektivě se musí commitovat, aby se objevily v perspektivě STORE a daly se ze strany klienta stáhnout

Dále je možno v rámci perspektivy BUFFER komponenty uploadovat.

1. **LIST** Resource objektů (buffer + store)
   1. Jednotlivé Resource zobrazeny v rozklikávacích blocích
   2. Struktura nadpisové řádku bude *symbolic name, version, presentation name, category*. Sloupce budou pokud možno graficky odděleny
   3. *Resource* se skládá z *Properties*, *Capabilities*, *Requirements* a *Categories*, každý tento oddíl má vlastní část bloku, dělené horizontálně
   4. Blok *Properties*
      1. Seznam jednotlivých *Properties*
      2. Struktura výpisu řádku: *Název, typ, hodnota* a tlačítko **EDIT**
      3. Na konci tlačítko **ADD NEW**
   5. Blok *Categories*
      1. Seznam kategorií, do kterých daná *Resource* zapadá, pod ním **EDIT**
   6. Blok *Capabilities*
      1. Vždy zobrazen název dané *Capability* a poté vypsány její *Properties*
      2. Struktura dílčích *Properties* stejná jako v prvním bloku, i tlačítko **EDIT**, na konci opět **ADD NEW**
   7. Blok *Resources*
      1. Struktura řádku: *Název, Filtr, Variables*
   8. Tlačítka **DELETE**, **DOWNLOAD, CHECK COMPATIBILITY** a checkbox pro pozdější EXECUTE
   9. Na konci výpisu komponent tlačítko **EXECUTE**
   10. Pouze v buffer části tlačítko **COMMIT**
   11. Pro výpis je možno aktivovat **FILTR**
2. Funkce **EDIT** (buffer + store)
   1. Editovat je možno každou *Property* buď přímo u *Resource* nebo u *Capabilit* a ost.
      1. Jeden typ formuláře – *Property* má stejnou strukturu
   2. Editace kategorií
      1. List katerogií a možnost přiřazení nebo odebrání
3. Funkce **UPLOAD** (buffer only)
   1. Tlačítko v hlavním menu
   2. Zobrazí formulář pro vložení souboru a potvrzení
   3. Komponenta se nauploadoje do *BUFFERU*
4. Funkce **COMMIT** (buffer only)
   1. Přesune *Resource* do seznamu *STORE*
5. Funkce **DOWNLOAD** (buffer + store)
   1. Umožní stažení \*.jar souboru s danou komponentou
6. Funkce **EXECUTE** (buffer + store)
   1. Na základě zaškrtnutých checkboxů u jednotlivých Resource vezme jejich seznam a přesune se na stránku s testováním, kde z tohoto seznamu bude vycházet
   2. Stránka s testováním obsahuje seznam testů, které je možno vstupním seznamu komponent spustit
   3. U jednotlivých testů bude možnost volby určitých parametrů
7. Funkce **CHECK COMPATIBILITY** (buffer + store)
   1. Po spuštění se odešle seznam *Requirements* dané *Resource* a ve výpisu *Requirementů* se zvýrazní ty, které neprošly kontrolou
8. Funkce **FILTR** (buffer + store)
   1. Omezení výpisu na základě zadání řetězce označujícího daný filtr
   2. Realizováno nejspíše vhodně označeným textovým polem na začátku obou perspektiv

## FÁZE II.

Vytvoření CoSi pluginu

1. Realizováno vytvořením nového indexeru
2. Po uploadu komponenty najít ve adresářové struktuře *Manifest* soubor, z toho si získat potřebná metadata a poté vytvořít *Resource* object
3. Je možno použí již existující *CoSiLoader*, který obsahuje funkční *ManifestReader* pro CoSi komponenty.

# Verze

|  |  |  |
| --- | --- | --- |
| Datum | Verze | Autor |
| 7.4.2011 | V 1.0 | Jan Techl |