1. **Popis použitých technologii**
2. **XML**

**eXtensible Markup Language - j**edná se o obecný značkovací jazyk, ve kterém mohou být popsány a definovány libovolné další formáty a data. Dokumenty ve formátu XML pak mohou být vytvářeny a zpracovávány v rozdílných aplikacích nebo informačních systémech. Jazyk je určen především pro výměnu dat mezi aplikacemi a pro publikování dokumentů, u kterých popisuje strukturu z hlediska věcného obsahu jednotlivých částí, nezabývá se vzhledem. Prezentace dokumentu (vzhled) může být definována pomocí CSS. Další možností zpracování je transformace pomoci XSL do jiného typu dokumentu, nebo do jiné aplikace XML. (viz. b)

1. **XSL**

**eXtensible Stylesheet Language** - pracuje na zcela jiném principu než kaskádové styly. XSL je jazyk, kterým vytváříme pravidla pro transformaci jedné třídy XML dokumentů na jiný XML dokument. Výsledný dokument, který vznikne aplikováním transformačních pravidel, je použit pro zobrazení. Dnes se nejčastěji XSL styl vytváří tak, aby výsledkem jeho aplikování byl HTML dokument. To umožňuje využít současná jádra prohlížečů, která si se zobrazením HTML poradí. Navíc lze XML dokumenty pomocí XSL transformovat do HTML ještě na serveru, a tak zpřístupnit XML dokumenty libovolnému prohlížeči.

1. **XSD**

**XML Schema Definition** - je jazyk, který popisuje strukturu XML dokumentu a kontroluje tak jeho správnost. Umožňuje mj. definovat povinné a volitelné XML elementy a atributy, datové typy a referenční integritu.

1. **Popis postupu**

Na základě zadání jsme přidali několik jednoduchých a jeden komplexní datový typ pro jednotlivé parametry produktů TT Klávesnice (SERVICE\_LIFE, KEY\_SPACING, MECHANICAL\_RESPONSE,KEY\_COUNT,INGRESS\_PROTECTION). Poslední z jmenovaných je kontrolován pomocí regulárního výrazu.  
  
Ukázkový dokument ukazuje několik produktů včetně nově definovaných parametrů.  
  
Pomoci XSL transformace jsme převedli zdrojové XML do HTML podoby a to velmi přímočaře. Jednoduše vypíše jejich hodnoty. Jedinou výjimkou je životnost, která je upravena tak, aby se zobrazovaly čísla ve formátu 10^6 apod. Počet cyklů se pohybuje ve statisících až milionech a tvar 1 000 000 je zbytečný. Vzhled vygenerovaného HTML dokumentu je definován pomocí CSS.

**Část CI:**  
  
metadata-input obsahuje plain-text naší práce, metadata-sample-settings cestu k tomuto souboru. Hlavní část je v transformaci metadata.xsl, kde jsou kromě dvou ukázkových typů informací přidány regexpy pro extrakci dalších. Vždy jde o element infotype. Extrahují se NACE kódy, adresa (nejsložitější regexp), HS kód, ISO norma a telefon. Struktura je variací na ukázkový element. Hlavní rozdíl spočívá v regexpech:

NACE je šest číslic, dvojtečka, bílé znaky a osmdesát znaků textu za tím, což by mělo obsáhnout textový popis kódu (přibližně).

Adresa je nejsložitější identifikuje adresy ve tvaru „Ulice, číslo, PSČ, město, stát“ oddělené čárkami nebo po řádcích. Má několik omezení, ale identifikuje obecným zápisem adresy v textu naší zprávy a i několik testovacích příkladů.

HS kód je složen ze čtyř číslic, nebo čtyř a dvou, nebo čtyř, dvou a čtyř. Vždy možno odděleno tečkami.

ISO norma může začínat řetězcem ČSN EN (evropská norma, tradiční systém při české standardizaci je tenhle prefix) následovaný ISO (případně ještě /IEC), jednou až pěti číslicemi a případně dvojtečkou s rokem vydání normy (na začátku číslice 1, nebo 2 a pak jakékoliv, takže rok 1000 by prošel, 2999 taky).  
  
Telefon je + a čtyři trojice číslic oddělené nepovinnou mezerou.