**Požadavek na změnu (RfC)[[1]](#endnote-2) – Z27656**

**a – věcné zadání**

# Základní informace

|  |  |
| --- | --- |
| **ID PK MZe**[[2]](#endnote-3)**:** | **508** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Název změny**[[3]](#endnote-4)**:** | **LPIS – Optimalizace aplikace předtisků – POC kontejnerizace pro kampaň 2020** | | | |
| **Datum předložení požadavku:** | | 15.7.2019 | **Požadované datum nasazení:** | 31.3.2020 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Kategorie změny**[[4]](#endnote-5)**:** | Normální  Urgentní | **Priorita**[[5]](#endnote-6)**:** | Vysoká  Střední  Nízká |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Oblas**t**:** | Aplikace | **Zkratka**[[6]](#endnote-7)**:** | LPIS | Verze: | 4.024.000021 |
| **Typ požadavku:** | Legislativní  Zlepšení  Reklamace  Bezpečnost | | |
| Infrastruktura | **Typ požadavku:** | Nová komponenta  Upgrade  Bezpečnost  Zlepšení  Obnova | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Role** | **Jméno** | **Organizace /útvar** | **Telefon** | **E-mail** |
|  |  |  |  |  |
| Metodický garant | Josef Miškovský | **SZIF** |  | Josef.Miskovsky@szif.cz |
| Žadatel | Oleg Blaško | Mze/11120 | 221814588 | [Oleg.blasko@mze.cz](mailto:Oleg.blasko@mze.cz) |
| Change koordinátor: | Jiří Bukovský | **CPR/11121** | 22181 2710 | Jiri.bukovsky@mze.cz |
| Poskytovatel / dodavatel: | xxx | **O2ITS** | xxx | xxx |

|  |  |  |  |
| --- | --- | --- | --- |
| **Smlouva č.**[[7]](#endnote-8)**:** | S2019-0043; DMS 391-2019-11150 | **KL:** | KL HR-001 |

# Stručný popis požadavku

## Popis a odůvodnění požadavku

Z hlediska technologického došlo v roce 2019 k přiblížení se k maximálnímu výkonovému potenciálu stávajícího řešení nastavením aplikačních serverů versus databáze LPIS a zejména u větších subjektů[[8]](#footnote-2) neexistuje již markantní prostor pro zrychlení aplikace. Současně s tímto technologickým problémem se ukazuje jako limitující faktor skutečnost, že kód předtiskové aplikace je součástí balíku aplikačního kódu nového LPIS vyvíjeného od roku 2014. Tento balík je již obtížně provozně spravovatelný a jako řešení se nabízí modernizace aplikačního řešení – tzv. **kontejnerizace**, kdy jednotlivé dílčí funkcionality budou osamostatněny ze stávajícího aplikačního balíku do tzv. **mikroslužeb** a budou provozovány na novém HW prostředí podporující spouštění mikroslužeb v kontejnerech.

Toto řešení umožní několik zásadních zlepšení:

* **Pro každou mikroslužbu bude možné škálovat výkon** (alokovat jí kapacity na serverech dle potřeby) – v průběhu předtisků není shodné využití jednotlivých funkcí – v době příprav předtisků je kladen důraz na kreslení, přípravy dat – hromadné deklarace, kdežto v poslední fázi kampaně jsou využívány masivně kopie dat, odesílání dat apod.
* U každé komponenty bude **možné provádět nasazení úprav nezávisle na nasazení kódu hlavního aplikačního balíku** – toto umožní výrazně zkrátit dobu od identifikace chyby po nasazení a zejména snížit čas kompilace a zjednodušit organizaci nasazení změn
* Přechodem na kontejnerizaci a **využití** **cache se sníží zátěž Oracle DB a budou méně výkonnostně ovlivňovány ostatní funkcionality LPIS**.

Je nutné zdůraznit, že změny **nemění uživatelské rozhraní ani business logiku**, ta zůstane pro uživatele totožná, bude se měnit technologické řešení mezi klientskou stranou, aplikačním serverem a databází a s tím související refaktoring zdrojových kódů.

Pro blížící se kampaň 2020 existuje několik komplikací:

* velké množství změnových požadavků upravujících stávající funkčnost nebo přidávající novou, což zvyšuje náročnost na čas implementace
* je složité předem určit HW nároky pro celé řešení a tím související nákladnost licencí pro potřebné SW vybavení
* kompletní upgrade předtisk aplikace znamená implementovat řadu mikroslužeb souvisejících s vlastními LPIS evidencemi, náročnost na čas implementace a HW nároky výsledného řešení rostou a hůře se odhadují
* na MZe není v současné době potřebný HW a SW

Po zvážení výše uvedených bodů je v rámci tohoto PZ navrženo implementovat pro kampaň 2020 proof of concept (POC) řešení, které **do mikroslužeb/kontejnerů a následného provozu přenese pouze vybranou funkčnost předtisků**. Na základě tohoto pilotního provozu bude možné zhodnotit výkonnost, upravit architekturu a návrh mikroslužeb celé Předtisk aplikace (či případných dalších mikroslužeb) a přesněji stanovit HW a SW nároky.

Současně bude zajištěno, že v případě provozních problémů navržené technologické změny bude možné zásahem dodavatele pomocí konfigurace přepnout dotčenou službu běžící v kontejnerovém řešení na aplikačně totožnou službu běžící na aplikačních serverech, která je součástí stávajícího programového balíku. Toto přepínání bude umožněno u všech dílčích služeb diverzifikovaných do kontejnerového řešení.

## Rizika nerealizace

V případě nerealizace může dojít k ještě vážnějším problémům při přípravě předtisků, než tomu bylo v roce 2019. V roce 2020 bude složitost předtisků větší o další povinnost rozšíření zákresů na straně jedné a o agendu přechodných závazků, která přípravu předtisků minimálně metodicky zkomplikuje. Další změny v roce 2021 povedou také k růstu složitosti. Z tohoto důvodu a kvůli eliminaci rizik je žádoucí přistoupit k navrženým krokům.

# Podrobný popis požadavku

Uživatelské rozhraní Předtisk aplikace je implementováno jako tzv. **SPA** (Single Page Application) pomocí JavaScriptu běžícího ve webovém prohlížeči a knihovny **Sencha ExtJS**. Prohlížeč se serverem komunikuje prostřednictvím vystaveného **REST** (Representational State Transfer) API ve formátu **JSON** (JavaScript Object Notation). Je využito tzv. **AJAX** (Asynchronous JavaScript and XML) volání, kdy při výměně dat mezi prohlížečem a serverem není třeba provést obnovu (refresh) celé stránky v prohlížeči. Tento koncept je moderní a není na něm třeba nic měnit.

Z pohledu uživatele však mají některé funkce Předtisk aplikace v uživatelském rozhraní pomalé odezvy trvající v extrémních případech minuty. Příčinou pomalosti jsou pomalé odezvy REST. Například pro vykreslení seznamu dílů uživatele s výměrami deklarovanými na opatření musí prohlížeč počkat na odpověď REST API, kdy na straně serveru je prováděna komplikovaná transformace množství dat z různých relačních tabulek databáze do podoby potřebné pro seznam. Navíc čím více má seznam položek, tím trvá transformace déle. Pokud trvá dlouho provedení REST, pak trvá dlouho i zobrazení seznamu.

Implementace REST API Předtisk aplikace je zabalena v tzv. hlavním aplikačním balíku, zdrojové kódy jsou implementovány jazykem Java kompatibilně s verzí JDK 1.7. Hlavní aplikační balík je provozován na JBoss EAP aplikačních serverech.

LPIS JBoss EAP servery

**Hlavní**

**aplikační**

**balík LPIS**

**Předtisky**

**JavaScript Single Page Application**

Portál interní

Eagri

**AJAX volání přes HTTPS**

**Předtisky**

**REST services**

Pro zrychlení aplikace je potřeba významně zrychlit odezvy REST API, což si vyžádá:

* úpravu architektury aplikace
* upgrade využitých technologií v souladu s navrženou architekturou
* změny ve zdrojových kódech REST služeb

Zároveň však při tomto upgrade musí zůstat zachovány beze změny položky:

* REST API (formát vstupů a výstupů)
* uživatelské rozhraní
* věcná funkčnost
* tabulky v relační databázi

Požadavky jsou následující:

1. Využít pro REST API předtisků novější serverové technologie, především novější verzi Java
2. Rozdělit implementaci na menší balíky, tzv. **mikroslužby** (microservices), které bude možné provozovat a testovat samostatně, bezstavově a oddělit je od ostatních aplikací, aby se navzájem výkonnostně neovlivňovaly
   1. Kódy budou vyčleněny ze stávajícího hlavního aplikačního balíku, do kterého jsou zabaleny další aplikace a jejichž technologický upgrade není předmětem tohoto PZ.
   2. Spolu s kódy REST API předtisků bude vyčleněna i funkčnost tříd z dalších vrstev, na kterých předtisky závisí, zejména metody pro získávání různých dat vázaných na DPB a souvisejících s opatřeními a ty budou opatřeny REST API pro interní užití.
   3. Dočasně lze ponechat REST API i v hlavním aplikačním balíku a pouze přesměrovat komunikaci na novou verzi, aby bylo případně možné přepnout komunikaci zpět
3. Provozovat mikroslužby a případný další podpůrný software v kontejnerech s vhodnou orchestrací, tzv. **kontejnerizační platforma**, aby bylo možné využívat výhod kontejnerizace: zajištění dostupnosti a výkonnosti pomocí více instancí a přiřazení prostředků (škálování), nasazování za chodu atp.
4. Doplnit do architektury dočasné úložiště tzv. **cache**, která umožní dočasně ukládat data v podobě, ve které jsou poskytována pro uživatelské rozhraní či interně pro dílčí operace.
5. Architekturu připravit tak, aby bylo možné stejným způsobem jako Předtisk aplikací upgradovat v budoucnosti i další aplikace a provozovat je na stejné kontejnerizační platformě
6. Používat stávající relační databázi Oracle RAC. Tabulky kvůli upgrade neměnit a nadále je využívat. V této databázi musí být uložena finální podoba dat předtisků, aby byla zachována konzistence, zpětná kompatibilita a tím zajištěna stabilita aplikací LPIS.
7. Při implementaci nové verze REST služeb využít produkty s následujícími vlastnostmi:
   1. jsou postavené na otevřené platformě (Open Source)
   2. jsou vhodné pro kontejnerizaci a podporované kontejnerizační platformou
   3. bude možné k nim případně zajistit podporu
8. Před uvedením do provozu bude provedeno testování Předtisk aplikace a ověření nové architektury:
   1. výkonnostní testy demonstrující řádové zrychlení u největších subjektů v ČR (zobrazení seznamů deklarací, založení sady, provedení kontrol opatření)
   2. ověření vysoké dostupnosti
   3. ověření škálování
   4. ověření kontinuálního nasazování (nasazování bez výpadku)
9. Upgrade Předtisk aplikace bude pilotním projektem a odrazovým můstkem pro upgrade a provoz dalších aplikací, po řádném otestování a odsouhlasení, že tato architektura a platforma splnila všechny požadavky, bude připraven plán upgrade dalších aplikací.

Pro splnění požadavků je v rámci upgrade Předtisk aplikace navrženo využít následující software:

* **Red Hat Openshift Container Platform (OCP)**
  + běh a orchestrace kontejnerů, v nichž budou umístěny mikroslužby pro Předtisk aplikaci a podpůrný software
  + vysoká dostupnost
  + zabezpečení
  + monitoring kontejnerů
  + logování dění v kontejnerech
  + kontinuální nasazování
  + škálování
  + tvoří cluster, kde vybrané servery jsou určeny pro řízení a správu a dalších N pro aplikační využití
  + bude využita komerční verze s podporou, kdy licence zahrnuje průběžný upgrade a podporu dalším produktům, zejména pro operační systém Red Hat Enterprise Linux, build OpenJDK a Red Hat SSO
  + licence je vázána na počet jader serverů určených pro aplikační využití
* **Spring Boot** 
  + pro implementaci mikroslužeb s REST API
  + dojde k refaktoringu zdrojových kódů pro přechod na Spring Boot platformu s využitím OpenJDK 11 bez EE (Enterprise Edition)
  + implementace práce s objektovým úložištěm
  + bude provozováno v kontejnerech
* **MongoDB**
  + zajišťuje objektové úložiště, které bude použito jako dočasné úložiště (cache) pro zrychlení odezev REST služeb
  + bude provozováno na samostatných serverech, s replikací - pro zajištění vysoké dostupnosti

Pro využití výhod kontejnerizace a využití cache dojde k refaktoringu předtiskových služeb a jejich rozdělení na menší provozované balíčky – mikroslužby. Výhod mikroslužeb lze využít v případě, že jsou z původního celku rozpadnuty s ohledem na to, co která REST služba zajišťuje. Mikroslužba poskytující vybranou předtiskovou funkčnost by měla mít následující vlastnosti:

1. malá
   1. malé množství implementovaného kódu zpřehlední a zrychlí vývoj, ladění a testování
   2. nasazení a start bude probíhat rychleji
2. bezstavová
   1. bude možné ji zastavit a opět spustit, aniž by došlo ke ztrátě dat
   2. umožní chod ve více instancích, aniž by musela být řešena replikace dat
3. vystavuje REST API
   1. jsou jasně definované vstupy a výstupy a chování vystavených metod (operací)
   2. při zásadní změně chování je možné udržovat více verzí API (verze je vedena v URL)
   3. dokumentaci je možné generovat přímo z kódu, využívá OpenAPI specifikaci
   4. lze implementovat integrační a výkonnostní testování
4. reprodukovatelnost
   1. pokud je již daná funkčnost implementována s definovaným rozhraním, pak je nežádoucí, aby se opakovala v jiné mikroslužbě
   2. lze vykonat její využití jinými službami a je vysoce žádoucí

Aplikační servery pro interní aplikace

Portál interní

Eagri

Aplikační servery pro farmářské aplikace

Aplikační servery pro úlohy na pozadí

Cluster určený pro kontejnery

Mapové servery

Oracle RAC

AgriBus

Aplikační servery WS

Mapové servery

Mapové servery

LDAP

**Ostatní servery**

Obrázek 1 Začlenění clusteru s kontejnerizační platformou

**Stávající LPIS servery**

**Nové LPIS servery**

**Předtisk služby**

**LPIS - schématické zobrazení začlenění clusteru pro kontejnery**

**Legenda:**

MongoDB

Hlavní balík

Hlavní balík

Obrázek 2 Začlenění clusteru s kontejnerizační platformou

Na Obrázku č.2 je znázorněno, že část předtiskových služeb bude provozována v kontejnerech a požadavky na tyto služby budou z portálů směřovány na ně. Ostatní služby zůstanou na současných aplikačních serverech a směřování zůstane stávající.

V rámci tohoto PZ bude do kontejnerů přenesena pilotně následující funkčnost:

1. **Načítání dat do seznamů deklarovaných dílů v sadě za kulturu** 
   1. na vstupu ID sady a kultury
   2. pokud budou výsledná data (JSON) připravena v MongoDB (dále cache), budou vrácena
   3. pokud nebudou výsledná data (JSON) připravena, bude zjištěna existence zdrojových dat v cache, pokud nebudou zdrojová data k dispozici, budou načtena z DB Oracle, z těchto dat a z již v MongoDB paměti připravených (nacachovaných) dat (například výsledky kontrol) budou sestavena výsledná data (JSON), uložena do cache a vrácena
   4. bude doplněna **nová služba pro zneplatnění dat v cache**, aby bylo možné vynutit opětovné sestavení
   5. bude doplněna **nová úloha běžící na pozadí** (také v kontejneru), která bude kontrolovat, zda nedošlo ke změně ve zdrojových datech (například v noci po nabytí účinnosti DPB) a která zneplatní zdrojová data v cache a tím vynutí opětovné sestavení
2. **Načítání dat do řádku seznamu deklarovaných dílů za kulturu a konkrétní díl** 
   1. na vstupu ID PTDPB
   2. obdoba 1) pouze omezena na daný díl
3. **Provedení kontrol na konkrétním dílu v sadě** 
   1. na vstupu ID PTDPB
   2. pokud jsou všechna podkladová data potřebná pro kontrolu připravena v MongoDB (cache), pak budou spuštěny všechny kontroly relevantní pro daný díl, výsledky uloženy do cache a vráceny na výstup
   3. pokud nebudou všechna podkladová data potřebná pro kontrolu připravena v cache, pak budou načtena z Oracle DB, uložena do cache a všechny kontroly relevantní pro daný díl, uloží výsledky do cache a vrátí
   4. bude doplněna nová služba pro zápis všech výsledků kontrol dílů do databáze Oracle za celou sadu a tato služba bude použita až před odesláním kontroly pro zafixování stavu
   5. bude doplněna **nová služba pro zneplatnění podkladových dat kontrol v cache**, aby bylo možné vynutit opětovné sestavení výsledků kontrol
   6. bude doplněna **nová úloha běžící na pozadí** (také v kontejneru), která bude kontrolovat, zda nedošlo ke změně v podkladových datech (například v noci po nabytí účinnosti ENVIRO či přepočtu překryvů atp.), která je zneplatní a tím vynutí opětovné sestavení
4. **Hromadné provedení kontrol dílu pro všechny díly v sadě**
   1. na vstupu ID SADY
   2. Spustí provedení 3) pro všechny díly v sadě
5. **Provedení kontrol opatření za danou sadu** 
   1. na vstupu ID SADY
   2. Obdoba 3) ale provede specifické hromadné kontroly opatření pro všechna relevantní opatření v sadě – pozor, nespouští kontroly všech dílů 4)
6. **Provedení všech kontrol** 
   1. na vstupu ID SADY
   2. Spustí 4) a 5)

Pro realizaci výše uvedených bodů bude navrženo, implementováno a dodáno následující:

* **mikroslužby** vystavující REST API a implementující asynchronní úlohy (využívající Spring Boot a MongoDB)
* **definice** **images** pro kontejnery
* generovaná **dokumentace REST API**
* **upravená SPA aplikace předtisků** tak, aby volala nová URL směřující na nové služby vystavené při implementaci výše uvedených funkčností, přičemž **bude možné konfiguračně určit, kterým uživatelům bude komunikace směřována na nové služby a kterým na původní služby (pro účely testování)**
* **konfigurace pro provoz implementovaných mikroslužeb** na kontejnerizační platformě (pods, services, routes, deployments atd)
* **související infrastruktura** (samotná kontejnerizační platforma, MongoDB)

## Dopad do zdrojových kódů

* využití Spring Boot znamená nutnou refaktorizaci (změna ve zdrojových kódech, beze změny logiky, ale je nutné důkladné otestování)
* využití asynchronnosti si žádá jiný přístup, algoritmizaci
* využití MongoDB (pro zmíněné využití jako cache) si žádá novou implementaci této práce s daty
* SQL pro nabírání dat z Oracle DB bude optimalizováno tak, aby nenabíralo data po konkrétních dílech (problematika velkých subjektů), ale hromadně za všechny DPB v sadě, popřípadě za celého uživatele

# Dopady na IS MZe

## Dopady na integraci s okolními systémy

* zdrojově jsou data načítána a cílově ukládána do centrální Oracle RAC databáze
* definice poskytovaných SOAP služeb pro AgriBus se v rámci tohoto PZ nijak nemění
* konzumace služeb využívaných z AgriBus se nijak nemění
* URL předtiskové aplikace na portále (ani specifikace využívaného REST API) se nijak nemění
* na portály bude potřeba dodat definice proxy na vybrané adresy směřující na kontejnerizační platformu

## Dopady na prostředí a hardware

Pro kontejnerizační platformu bude nutné dodat hardware poplatný nárokům pro chod kontejnerizační platformy (clusteru) a dále hardware pro chod kontejnerů s předtiskovými službami. Z důvodu úspory hardware **bude v rámci pilotního řešení provozován v prostředí MZE jeden cluster**, který bude obsahovat **kontejnery pro provoz a předprovozní testování nové release**. **Cluster pro vývoj a interní testování si zajistí dodavatel vlastní, mimo prostředí MZE.**

Prostředí může být v rámci clusteru realizováno více než jedno, znamená vlastně sadu nasazených mikroslužeb dané verze. Lze tedy mít spuštěno kontejnery s různými verzemi a realizovat tak více prostředí – v souladu s napojením na adekvátní okolní systémy. Množství je omezeno dostupnými prostředky serverů (paměť, procesory, disky), nasazování více verzí/prostředí pak je o více nasazovacích linkách.

Navržené clustery a prostředí:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cluster | HA clusteru | Prostředí / verze | Přístup přes portály | Databáze | HA služeb |
| Pilotní (MZE) | Ano | Provozní | eagri.cz  portal.mze.cz | Provozní (MZE) | Ano |
| Předprovozní *– verze shodná s provozní po nasazení či verze o jedno vyšší před nasazením nové release* | eagritest.cz  portaltest.mze.cz | Testovací (MZE) | Ne |
| Vývojový  (dodavatel) | Ano | Testovací - *verze vyšší než provozní, určené pro interní testování další release* | - | Vývojová (dodavatel) | Ano |
| Vývojové – *verze vyšší než testovací, určené pro vývojáře* | - | Vývojová (dodavatel) | Ne |

Mikroslužbu v rámci kontejneru je možné nasazovat samostatně, kdy nasazení opravy či změny dílčí funkčnosti nemá dopad na chod celého systému. Lze zajistit bezvýpadkové nasazení zvýšením počtu kontejnerů v době nasazení a po úspěšném naběhnutí nové verze a přesměrování komunikace původní verzi vypnout.

**Vysoká dostupnost (HA) clusteru** vyžaduje zajištění dostupnosti komponent nutných pro chod clusteru/platformy, což znamená provoz minimálně tři tzv. master nodes.

**Vysoká dostupnost (HA) služeb** vyžaduje zajištění dostupnosti služeb, což znamená spuštění minimálně dvojnásobného množství kontejnerů dané verze.

Sumární nároky na HW jsou uvedeny v následující tabulce:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| VM pro | Počet VM | CPU fyzických celkem | RAM celkem | Disk celkem | Poznámka |
| OCP – master nodes, včetně etcd | 3 | 6 | 48 | 300 | dle <https://docs.openshift.com/container-platform/3.11/install/prerequisites.html>  -  minimální nároky na master node **4 vCPU a 16GB RAM a kvůli HA je u etcd potřeba mít minimálně 3 nodes**  **- pro toto není nutná licence OCP, licencují se jádra aplikačních nodes** |
| OCP – infra nodes: monitoring, repository, atp | 2 | 4 | 32 | 400 | <https://docs.openshift.com/container-platform/3.11/install/prerequisites.html> - nodes zapojené do clusteru , kde běží repository, nástroje pro správu, monitoring atp **- pro toto není nutná licence OCP, licencují se jádra aplikačních nodes** |
| OCP – aplikační nodes | 4 | 8 | 64 | 400 | - **je třeba zajistit licence OCP pro 8 fyzických jader**, tedy v případě, že procesor podporuje hyperthreading, pak 8 jádrová OCP subscription umožňuje provoz 16 vCPU |
| MongoDB – noSQL databáze | 3 | 6 | 60 | 2250 | db cluster – replikace master/slave  <https://docs.mongodb.com/manual/replication/>, **je třeba mít minimálně 3 nodes** |
|  |  | **24** | **204** | 3350 |  |

**Je žádoucí a doporučené, aby fyzická CPU podporovala hyperthreading, přepočet je pak vCPU = 2\*CPU, přičemž není třeba dokupovat další licence OCP.**

**Licence ve výše uvedeném rozsahu zajistí MZe v rámci součinnosti.**

## Požadavky na součinnost Agribus

V rámci tohoto pilotního řešení není třeba žádná součinnost s AgriBus.

## Dotčené konfigurační položky[[9]](#endnote-9)

|  |  |  |
| --- | --- | --- |
| **ID** | **Název položky** | **Předpokládaný dopad** |
| 7 | n2rhpvn3.apl.mzem.net | Nasazení nové verze aplikace |
| 8 | n2rhpvn4.apl.mzem.net | Nasazení nové verze aplikace |
| 9 | n2rhpvq1.apl.mzem.net | Nasazení nové verze aplikace |
| 10 | n2rhpvq2.apl.mzem.net | Nasazení nové verze aplikace |

Dále budou dotčeny nové konfigurační položky, které plynou z nasazení řešení kontejnerizace (viz obrázek č. 2 a kapitola 4.2.)

## Bezpečnost

PZ je nezbytné vyvíjet s ohledem na Směrnici standardu systémové bezpečnosti 2.4, přičemž charakter úprav neznamená žádná rizika z hlediska bezpečnosti.

## Rizika implementace změny

Rizikem je časové nezvládnutí implementace kontejnerizace aplikačního řešení LPIS. Je nezbytné vyvíjet aplikaci tak, aby bylo možné použít dosavadní řešení a ad hoc dílčí funkcionality přepínat na nové služby.

## Požadavek na podporu provozu naimplementované změny

Podpora bude zajištěna v rámci stávajícího katalogového listu LPIS -001 a LPIS-002 v souladu s podmínkami smlouvy PRAIS bez navýšení ceny za paušální služby.

# Požadavek na dokumentaci[[10]](#endnote-10)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Dokument** | **Formát výstupu** (ano/ne) | | |
| el. úložiště | papír | CD |
|  | Analýza navrhnutého řešení – implementační dokument | ANO | NE | NE |
|  | Dokumentace dle specifikace Závazná metodika návrhu a dokumentace architektury MZe[[11]](#endnote-11) | ANO | NE | NE |
|  | Testovací scénář, protokol o otestování | ANO | ANO | NE |
|  | Uživatelská příručka | ANO | NE | NE |
|  | Provozně – bezpečnostní dokumentace | ANO | NE | NE |
|  | Zdrojový kód a měněné konfigurační soubory (průběžně paralelně na základě pravidelných aktualizací) včetně příslušné dokumentace dle „best practice“ | ANO | NE | NE |
|  | Dojde- li ke změně chování webových služeb.WS technická dokumentace - WS – aktualizace a doplnění dokumentace dotčených webových služeb (WSDL, povolené hodnoty včetně popisu významu, případně odkazy na externí číselníky, vnitřní logika služby, chybové kódy s popisem, popis logování na úrovni služby) | ANO | NE | NE |
|  | Kontejnery – obsahující minimálně seznam kontejnerů, jejich obsah (co obsahuje image, včetně licenčního modelu), popis způsob přeložení a způsob (skripty) pro nasazení, včetně příslušného popisu dle „best practice“, popis způsobu přenosu přeloženého kontejneru do prostředí, popis kontejener nebo image repository, atd. | ANO | NE | NE |
|  | Konfigurace a dokumentace kontejnerizace – detailní popis prostředí a jeho konfigurace, včetně integrací, whitelist pro proxy, prostupy, komunikační mapa, atd. | ANO | NE | NE |

(**ROZSAH TECHNICKÉ DOKUMENTACE**

1. **Sparx EA modelu (zejména ArchiMate modelu)**

V případě, že v rámci implementace dojde k změnám architektury, provede se aktualizace modelu. Sparx EA model by měl zahrnovat:

* 1. aplikační komponenty tvořící řešení, případně dílčí komponenty v podobě ArchiMate Application Component,
  2. vymezení relevantních dílčích funkcionalit jako ArchiMate koncepty, Application Function přidělené k příslušné aplikační komponentě (Application Component),
  3. prvky webových služeb reprezentované ArchiMate Application Service,
  4. hlavní datové objekty a číselníky reprezentovány ArchiMate Data Object,
  5. activity model/diagramy anebo sekvenční model/diagramy logiky zpracování definovaných typů dokumentů,
  6. popis použitých rolí v systému a jejich navázání na související funkcionality (uživatelské role ve formě ArchiMate konceptu Data Object a využití rolí v rámci funkcionalit/ Application Function vazbou ArchiMate Access),
  7. doplnění modelu o integrace na externí systémy (konzumace integračních funkcionalit, služeb a rozhraní), znázorněné ArchiMate vazbou Used by.

1. **Bezpečnostní dokumentace**

Jde o přehled bezpečnostních opatření, který jen odkazuje, kde v technické dokumentaci se nalézá jejich popis

Jedná se především o popis těchto bezpečnostních opatření (jsou-li relevantní):

* 1. řízení přístupu, role, autentizace a autorizace, druhy a správa účtů,
  2. omezení oprávnění (princip minimálních oprávnění),
  3. proces řízení účtů (přidělování/odebírání, vytváření/rušení),
  4. auditní mechanismy, napojení na SIEM (Syslog, SNP TRAP, Textový soubor, JDBC, Microsoft Event Log…),
  5. šifrování,
  6. zabezpečení webového rozhraní, je-li součástí systému,
  7. certifikační autority a PKI,
  8. zajištění integrity dat,
  9. zajištění dostupnosti dat (redundance, cluster, HA…),
  10. zálohování, způsob, rozvrh,
  11. obnovení ze zálohy (DRP) včetně předpokládané doby obnovy,
  12. předpokládá se, že existuje síťové schéma, komunikační schéma a zdrojový kód.

# Akceptační kritéria

Plnění v rámci požadavku na změnu bude akceptováno, jestliže budou akceptovány dokumenty uvedené v tabulce výše v bodu 5 a budou předloženy protokoly o uživatelském testování podepsané garantem, který je uveden ve sloupci Akceptuje.

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Akceptační kritérium** | **Způsob verifikace** | **Akceptuje** |
|  | Fungování nových úprav | Testovací scénáře | odborní garanti |
|  | Předložení dokumentace | Dokumentace | odborní garanti + change koordinátor |

# Základní milníky

|  |  |
| --- | --- |
| **Milník** | **Termín** |
| Nasazení na testovací prostředí | 15.3.2020 |
| Nasazení na provozní prostředí | 31.3.2020 |
| Dodání dokumentace | 30.4.2020 |
| Akceptace | 31.5.2020 |

# Přílohy

1.

2.

# Podpisová doložka

|  |  |  |  |
| --- | --- | --- | --- |
| **Za resort Mze:** | **Jméno:** | **Datum:** | **Podpis:** |
| Metodický garant | Josef Miškovský |  |  |
| Žadatel | Lenka Typoltová |  |  |
| Change koordinátor: | Jiří Bukovský |  |  |

**B – nabídkA řešení k požadavku Z27656**

|  |  |
| --- | --- |
| **ID PK MZe**: | 508 |

# Návrh konceptu technického řešení

Viz část A tohoto PZ, body 2 a 3.

**Nezbytným předpokladem pro realizaci tohoto PZ je vytvoření prostředí na straně MZe, kde poběží jak samotná platforma, tak i technologie, které budou součástí řešení.**

# Uživatelské a licenční zajištění pro Objednatele

V souladu s podmínkami smlouvy 391-2019-11150

# Dopady do systémů MZe

(Pozn.: V popisu dopadů zohledněte strukturu informací uvedenou v části A - Věcné zadání v bodu 4. U, přičemž u dopadů dle bodu 4.1 uveďte, zda může mít změna dopad do agendy, aplikace, na data, na síťovou strukturu, na serverovou infrastrukturu, na bezpečnost.)

# Dopady do agendy

# bez dopaduDopady na aplikace

Dopady viz kapitola 4.1 – integrace na ostatní systémy

# Dopady na data bez dopadu

# Dopady na serverovou infrastrukturu Dopady viz. kapitola 4.2 – dopady na prostředí a HW

# Dopady na dohledové scénáře[[12]](#endnote-12) bez dopad

# Dopady na bezpečnost bez dopadu

Návrh řešení musí být v souladu se všemi požadavky v aktuální verzi Směrnice systémové bezpečnosti MZe. Upřesnění požadavků směrnice ve vztahu k tomuto RfC:

|  |  |  |
| --- | --- | --- |
| **Č.** | **Oblast požadavku[[13]](#endnote-13)** | **Předpokládaný dopad a navrhované opatření/změny** |
|  | Řízení přístupu 3.1.1. – 3.1.6. | Bez dopadu |
|  | Dohledatelnost provedených změn v datech 3.1.7. | Bez dopadu |
|  | Centrální logování událostí v systému 3.1.7. | Bez dopadu |
|  | Šifrování 3.1.8., Certifikační autority a PKI 3.1.9. | Bez dopadu |
|  | Integrita – constraints, cizí klíče apod. 3.2. | Bez dopadu |
|  | Integrita – platnost dat 3.2. | Bez dopadu |
|  | Integrita - kontrola na vstupní data formulářů 3.2. | Bez dopadu |
|  | Ošetření výjimek běhu, chyby a hlášení 3.4.3. | Bez dopadu |
|  | Práce s pamětí 3.4.4. | Bez dopadu |
|  | Řízení - konfigurace změn 3.4.5. | Bez dopadu |
|  | Ochrana systému 3.4.7. | Bez dopadu |
|  | Testování systému 3.4.9. | Bez dopadu |
|  | Externí komunikace 3.4.11. | Bez dopadu |

# Dopady na síťovou infrastrukturu

(Pozn.: V případě, že má změna dopady na síťovou infrastrukturu, doplňte tabulku v připojeném souboru - otevřete dvojklikem.)

# Ostatní dopady

(Pozn.: Pokud má požadavek dopady do dalších požadavků MZe, uveďte je také v tomto bodu.)

# Požadavky na součinnost Objednatele a třetích stran

|  |  |
| --- | --- |
| **MZe / Třetí strana** | **Popis požadavku na součinnost** |
| MZe | Realizace tohoto PZ předpokládá vytvoření prostředí na MZe, kde poběží jak samotná platforma, tak i technologie, které budou součástí řešení. |
| MZe | Součinnost při testování a akceptaci PZ |

(Pozn.: K popisu požadavku uveďte etapu, kdy bude součinnost vyžadována.)

# Harmonogram plnění[[14]](#endnote-14)

|  |  |
| --- | --- |
| **Milník** | **Termín \*/** |
| Nasazení na provozní prostředí | 31.3.2020 |
| Dodání dokumentace | 30.4.2020 |
| Akceptace | 31.5.2020 |

\*/ Upozornění: Uvedený harmonogram je platný v případě, že Dodavatel obdrží objednávku v rozmezí 13.2.-20.2.2020. V případě pozdějšího data objednání si Dodavatel vyhrazuje právo na úpravu harmonogramu v závislosti na aktuálním vytížení kapacit daného realizačního týmu Dodavatele či stanovení priorit ze strany Objednatele.

# Pracnost a cenová nabídka navrhovaného řešení

včetně vymezení počtu člověkodnů nebo jejich částí, které na provedení poptávaného plnění budou spotřebovány

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Oblast / role**[[15]](#endnote-15) | **Popis** | **Pracnost v MD/MJ** | **v Kč bez DPH** | **v Kč s DPH** |
|  |  |  |  |  |
|  | Viz cenová nabídka v příloze č.01 | 268,75 | 2 391 875,00 | 2 894 168,75 |
| **Celkem:** | | 268,75 | 2 391 875,00 | 2 894 168,75 |

(Pozn.: MD – člověkoden, MJ – měrná jednotka, např. počet kusů)

Případné další informace.

# Přílohy

|  |  |  |
| --- | --- | --- |
| **ID** | **Název přílohy** | **Formát**  **(CD, listinná forma)** |
| 01 | Cenová nabídka | Listinná forma |
| 02 | Detailní rozpad | E-mailem |

# Podpisová doložka

|  |  |  |  |
| --- | --- | --- | --- |
| **Název Dodavatele / Poskytovatele** | **Jméno** **oprávněné osoby**[[16]](#endnote-16) | **Datum** | **Podpis** |
| O2 IT Services s.r.o. | xxx | 14.2.2020 |  |

**C – Schválení realizace požadavku Z27656**

|  |  |
| --- | --- |
| **ID PK MZe**: | 508 |

# Specifikace plnění

Požadované plnění je specifikováno v části A a B tohoto RfC.

Dle části B bod jsou pro realizaci příslušných bezpečnostních opatření požadovány následující změny[[17]](#footnote-3):

# Uživatelské a licenční zajištění pro Objednatele (je-li relevantní):

|  |  |  |
| --- | --- | --- |
| **Č.** | **Oblast požadavku[[18]](#endnote-17)** | **Předpokládaný dopad a navrhované opatření/změny** |
|  | Řízení přístupu 3.1.1. – 3.1.6. | Bez dopadu |
|  | Dohledatelnost provedených změn v datech 3.1.7. | Bez dopadu |
|  | Centrální logování událostí v systému 3.1.7. | Bez dopadu |
|  | Šifrování 3.1.8., Certifikační autority a PKI 3.1.9. | Bez dopadu |
|  | Integrita – constraints, cizí klíče apod. 3.2. | Bez dopadu |
|  | Integrita – platnost dat 3.2. | Bez dopadu |
|  | Integrita - kontrola na vstupní data formulářů 3.2. | Bez dopadu |
|  | Ošetření výjimek běhu, chyby a hlášení 3.4.3. | Bez dopadu |
|  | Práce s pamětí 3.4.4. | Bez dopadu |
|  | Řízení - konfigurace změn 3.4.5. | Bez dopadu |
|  | Ochrana systému 3.4.7. | Bez dopadu |
|  | Testování systému 3.4.9. | Bez dopadu |
|  | Externí komunikace 3.4.11. | Bez dopadu |

# Požadavek na součinnost

|  |  |  |
| --- | --- | --- |
| **Útvar / Dodavatel** | **Popis požadavku na součinnost** | **Odpovědná osoba** |
| MZe | Realizace tohoto PZ předpokládá vytvoření prostředí na MZe, kde poběží jak samotná platforma, tak i technologie, které budou součástí řešení. |  |
| MZe | Součinnost při testování a akceptaci PZ | Garanti MZe |

# Harmonogram realizace[[19]](#endnote-18)

|  |  |
| --- | --- |
| **Popis etapy** | **Termín** |
| Nasazení na provozní prostředí | 31.3.2020 |
| Dodání dokumentace | 30.4.2020 |
| Akceptace | 31.5.2020 |

\*/ Upozornění: Uvedený harmonogram je platný v případě, že Dodavatel obdrží objednávku v rozmezí 13.2.-20.2.2020. V případě pozdějšího data objednání si Dodavatel vyhrazuje právo na úpravu harmonogramu v závislosti na aktuálním vytížení kapacit daného realizačního týmu Dodavatele či stanovení priorit ze strany Objednatele.

# Pracnost a cenová nabídka navrhovaného řešení

včetně vymezení počtu člověkodnů nebo jejich částí, které na provedení poptávaného plnění budou spotřebovány

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Oblast / role**[[20]](#endnote-19) | **Popis** | **Pracnost v MD/MJ** | **v Kč bez DPH:** | **v Kč s DPH:** |
|  |  |  |  |  |
|  | Viz cenová nabídka v příloze č.01 | 268,75 | 2 391 875,00 | 2 894 168,75 |
| **Celkem:** | | 268,75 | 2 391 875,00 | 2 894 168,75 |

(Pozn.: MD – člověkoden, MJ – měrná jednotka, např. počet kusů)

# Případné další obchodní podmínky[[21]](#endnote-20)

# Posouzení[[22]](#endnote-21)

|  |  |  |  |
| --- | --- | --- | --- |
| **Role** | **Jméno** | **Datum** | **Podpis/Mail[[23]](#endnote-22)** |
| Bezpečnostní garant | Oldřich Štěpánek | 14.2.2020 | Viz. Příloha 2 |
| Provozní garant | Štětina Pavel | 14.2.2020 | Viz. Příloha 3 |
| Architekt |  |  |  |

# Schválení

|  |  |  |  |
| --- | --- | --- | --- |
| **Za resort Mze:** | **Jméno:** | **Datum:** | **Podpis:** |
| Metodický/Věcný garant | Josef Miškovský |  |  |
| Žadatel | Oleg Blaško |  |  |
| Change koordinátor: | Jiří Bukovský |  |  |
| Oprávněná osoba dle smlouvy | Vladimír Velas |  |  |

# Vysvětlivky

1. Formulář RfC je tvořen třemi částmi, A - Věcné zadání, B – Nabídka řešení, C - Potvrzení realizace požadavku. První část (Věcné zadání) je předložena poskytovateli/dodavateli jako pobídka k předložení nabídky řešení. Druhou část, tj. část B použije dodavatel řešení k vypracování nabídky, kterou předloží MZe. Třetí část (Potvrzení realizace požadavku) se po vyplnění přiloží k první a druhé části a předloží se ke schválení osobám uvedeným v části C RfC. Poskytovateli/dodavateli se poté vyplněný formulář RfC předkládá v příloze objednávky na realizaci změnového požadavku. Pouze tato podepsaná objednávka je pokynem pro dodavatele/poskytovatele k realizaci změny. [↑](#endnote-ref-2)
2. ID PK MZe – pomocný identifikátor požadavku přidělený v pomocné evidenci projektové kanceláře MZe [↑](#endnote-ref-3)
3. Předmět změny – stručná informace, název požadavku [↑](#endnote-ref-4)
4. Kategorie změny – kategorie urgentní se využije v naléhavých případech, kdy je třeba vyřešit nedostupnost zásadní funkcionality systému vzhledem ke zpracování agendy, pro jejíž podporu systém slouží. [↑](#endnote-ref-5)
5. Priorita – vyjadřuje důležitost zapracování požadavku z pohledu časového. Vyplní se v případě volby kategorie „Normální změna“. [↑](#endnote-ref-6)
6. Zkratka – zkratka aplikace (viz „kód služby“ v katalogu služeb) [↑](#endnote-ref-7)
7. Smlouva č. – uvede se, pokud existuje smlouva, v rámci níž se požadavky předkládají, totéž platí pro KL (katalogový list). [↑](#endnote-ref-8)
8. Za velký subjekt se považuje subjekt s min. 300 DPB (dílů půdních bloků) [↑](#footnote-ref-2)
9. Vyplňte ve spolupráci s provozním garantem. [↑](#endnote-ref-9)
10. Vyplní Change koordinátor s Provozním garantem. Uvedený seznam dokumentace je pouze příkladem. [↑](#endnote-ref-10)
11. Rozsah požadované dokumentace uveďte do tabulky. [↑](#endnote-ref-11)
12. Pokud z vyhodnocení dopadů vyplyne potřeba upravit dohledové scénáře nebo zpracování nového scénáře, pak se má za to, že položka seznamu „Požadavek na dokumentaci“ v b. 5 části A RfC „Dohledové scénáře (úprava stávajících/nové scénáře)“ je vyžadována a bude součástí akceptačního řízení, nebude-li v části C RfC v bodu 1 „Specifikace plnění“ stanoveno jinak. [↑](#endnote-ref-12)
13. Jednotlivé oblasti – položky v tabulce korespondují s kapitolami Standardu systémové bezpečnosti. [↑](#endnote-ref-13)
14. Uvede se datum zahájení a ukončení realizace, příp. další etapy. [↑](#endnote-ref-14)
15. Role se vyplní pouze v relevantních případech, např. u požadavku na infrastrukturu. [↑](#endnote-ref-15)
16. Oprávněná osoba – smluvně určená osoba oprávněná k předkládání požadavku na předložení nabídky. [↑](#endnote-ref-16)
17. Potvrzení realizace příslušných opatření/změn vyznačí posuzovatel za Oddělení kybernetické bezpečnosti. [↑](#footnote-ref-3)
18. Jednotlivé oblasti – položky v tabulce korespondují s kapitolami Standardu systémové bezpečnosti. [↑](#endnote-ref-17)
19. Uvede se datum zahájení a ukončení realizace, příp. další etapy. [↑](#endnote-ref-18)
20. Role se vyplní pouze v relevantních případech, např. u požadavku na infrastrukturu. [↑](#endnote-ref-19)
21. Změna smluvních podmínek - vyplní se v případě, že dohodnuté podmínky realizace požadavku se liší od smluvních. [↑](#endnote-ref-20)
22. RfC se zpravidla předkládá k posouzení Bezpečnostnímu garantovi, Provoznímu garantovi, Architektovi, a to podle předpokládaných dopadů změnového požadavku na bezpečnost, provoz, příp. architekturu. Change koordinátor rozhodne, od koho vyžádat posouzení dle konkrétního případu změnového požadavku. [↑](#endnote-ref-21)
23. Doplní se podpis nebo se uvede odkaz na mailovou zprávu, v které bylo posouzení doručeno. [↑](#endnote-ref-22)