ČNB ČESKÁ NÁRODNÍ BANKA

Věcné zadání projektu SDAT

F - Uživatelé a přístupová práva

Červenec 2016

Obsah

1 Úvod

Účelem dokumentu je popsat všechny možné přístupy uživatelů, ať už registrovaných nebo i neregistrovaných k systému SDAT a způsob, jakým těmto uživatelům budou přidělována a vyhodnocována přístupová práva.

Systém SDAT vychází primárně z předpokladu, že k systému přistupuje uživatel, který je v systému registrován a takový uživatel nejdříve projde autentizačním procesem. Na základě znalosti byznys požadavků je však třeba zajistit přístup i tzv. neregistrovanému uživateli, tedy uživateli, který nebude prokazovat svoji identitu pomocí autentizačního procesu, ale pouhým jednorázovým zadáním několika základních osobních údajů.

Protože níže uvedené procesy pracují s objekty Osoba a Uživatel, je účelem tohoto dokumentu taktéž kompletní **popis objektového modelu pro správu uživatelů a oprávnění.**

Protože dokument řeší proces vytvoření uživatelského účtu, pomocí kterého se bude uživatel dále před systémem identifikovat (autentizovat), je účelem tohoto dokumentu **taktéž popis bezpečnostní politiky,** která s procesem autentifikace bezprostředně souvisí**.**

V dalším textu jsou uživatelé členěni na interní a externí. V případě, že není zcela explicitně uvedeno, zda se jedná o externího nebo interního uživatele, lze text chápat tak, že platí pro jakéhokoli uživatele, bez dalšího členění. Dále platí, že pokud se mluví o uživateli, má se za to, že se jedná o uživatele, který má právo provést danou akci/aktivitu. V případě, že je v textu zmíněn pojem „administrátor", má se za to, že se jedná o standardního uživatele, který je přiřazen k uživatelskému místu, ke kterému jsou navázané takové role/aktivity, které uživateli umožní vykonávat administrátorské aktivity.

Předmětem tohoto dokumentu je popis těchto procesů:

- **proces v ČNB řízeného založení aplikačního účtu uživatele** – proces, kdy Osobu/aplikační účet uživatele zakládáme v ČNB a potřebujeme bezpečným způsobem předat uživateli přihlašovací informace (credentials),
- **proces vytvoření externí Osoby a aplikačního účtu uživatele externím Osobou (Autoregistrace)** - proces, kdy si Osoba, které vznikla Vykazovací povinnost, sama založí záznam do Registru osob a vytvoří si aplikační účet uživatele, pomocí kterého splní svoji Vykazovací povinnost. Tento proces bude využíván i v případě, že ČNB bude potřebovat provést šetření u velkého počtu Osob, které v danou chvíli neexistují v Registru osob a jejich založení v SDAT by nebylo efektivní (jedná se o jednorázové šetření nebo dobrovolnou účast). Systém autoregistrace umožní těm, kteří se šetření chtějí zúčastnit, zaregistrovat se k systému SDAT a předat ČNB požadovaná data. Autoregistrace tak umožní delegovat na zúčastněné Osoby proces, který by bylo neefektivní vykonávat centrálně z ČNB,
- **proces přístup neregistrovaného uživatele** – proces bude použit v případě, kdy Osoba chce pomocí systému SDAT zaslat data, aniž by jí musel být aplikační účet uživatele vytvořen v ČNB. Tento proces umožní integrovat do SDAT vykazovací povinnosti, které jsou v současné době realizovány systémem SIPRES.

Součástí výše zmíněných procesů je popis některých dalších subprocesů, např. procesu přístupu uživatele k systému v případě zapomenutého hesla, případně proces rozšířeného ověření identity uživatele pomocí dvoukrokové autentifikace.

Pro pochopení dalšího textu je třeba si definovat základní pojmy pro oblast ověřování identity uživatele a definice a vyhodnocování přístupových práv:

- **autentifikace** - proces ověření identity uživatele. Tento proces je prováděn nejčastěji zadáním správné kombinace uživatelského jména a hesla. Existují případy, kdy je autentifikace prováděna jinými způsoby. Viz pojmy SSO a Dvoukroková autentifikace,
- **SSO** - Single Sign On. Speciální druh autentifikace, kdy k prokázání identity uživatele není třeba znalost jména a hesla, ale identita uživatele se přebírá z operačního systému. Předpokládá se, že operační systém je schopen identitu ověřit a jakmile tato identita je jednou ověřena, není potřeba ji v dalším systému, tedy SDAT, znovu prokazovat. Tento způsob autentifikace bude SDAT používat u interních uživatelů (ČNB), nelze jej však použít pro externí uživatele,
- **dvoukroková (dvoufaktorová) autentizace** - způsob ověření identity uživatele, kdy k jejímu prokázání je třeba kromě znalosti jména a hesla použit ještě další údaj. V případě aplikace SDAT bude tímto údajem jednorázový PIN zaslaný během procesu prokazování identity uživatele na jeho mobilní telefon. Tento způsob autentifikace bude v některých případech (založení aplikačního účtu uživatele autoregistrací) použit pouze pro ověřování identity externích uživatelů,
- **oprávnění** - vymezuje jaké aktivity a s jakými daty smí uživatel v systému provádět. Oprávnění uživatele je složeno z definice přístupových práv a rozsahu oprávnění,
- **přístupová práva** - určení toho, CO uživatel v systému smí dělat. Přístupová práva znamenají definici povolených aktivit. Určují tedy například, že uživatel má povolenou aktivitu „Vytvořit číselník“, ale například nemá povolenou aktivitu „Vytvořit Účet“,
- **rozsah oprávnění** - určení toho, S ČÍM (= S JAKÝMI DATY) smí uživatel provádět aktivity, které má povoleny na základě nastavení přístupových práv. Tedy například pokud uživatel má nastaveno přístupové právo „Čtení hodnot údajů“, pak rozsah oprávnění definuje, pro jaké Osoby toto přístupové právo platí. Toto právo může platit pro všechny Osoby (pak má uživatel právo číst hodnoty údajů za všechny Osoby, které SDAT eviduje) anebo může být nastaveno pouze pro vyjmenované Osoby, například „A“ a „B“. V takovém případě rozsah oprávnění omezuje přístupové právo uživatele – nadále sice má povoleno „číst hodnoty údaje“, systém mu umožní číst hodnoty údajů pouze pro Osoby „A“ a „B“, ale zamezí přístupu k hodnotám údajů Osoby „C“. Detailně je tento objekt popsán v kapitole 2.9 Objekt Rozsah oprávnění.

2 Objektový model

2.1 Základní pravidla

Následující text vymezuje základní pravidla, ze kterých bylo vycházeno při tvorbě objektového modelu pro oblast uživatelů a přístupových práv:

- data v aplikaci SDAT se dělí (pro účely řízení přístupových práv) **na „metadata“ a „data“.** Z hlediska řízení přístupových práv to znamená:
  - v rámci „metadat“ **je třeba řídit přístupy k projektovaným Výkazům.** Je tedy třeba určit, jaké Výkazy může uživatel upravovat (projektovat) a jaké ne. Pokud se hovoří o řízení přístupových práv na metadata, nemyslí se tím přístup řízený přes instance všech objektů metapopisu, ale pouze na instance objektu Výkaz. Řízení přístupových

práv na metadata se navíc týká pouze procesu projektování Výkazů. Jakékoli čtení z objektů metapopisu není nutné jakkoli řídit a to ani pro interní, ani pro externí uživatele, protože celá oblast metapopisu je veřejná (i externě mimo ČNB),
  - v rámci „dat“ **je třeba řídit přístupy ke čtení hodnot vykázaných údajů.** To znamená, že pokud interní uživatel má pracovat s hodnotami zaslanými Osobami, je třeba určit, s JAKÝMI hodnotami může pracovat. To znamená, že je třeba určit:
    - s daty za jakou Osobu,
    - s daty za jaké Výkazy,
    - s jakou úrovní citlivostí dat (citlivé vs. necitlivé údaje) smí uživatel pracovat.
- uživatelé se dělí **na „interní“ a „externí“:**
  - **interním uživatelem** je zaměstnanec ČNB. Jen tento typ uživatelů může, v závislosti na definici přístupových práv, provádět úpravy metapopisu, tedy „metadat“. Interní uživatel může přistupovat kromě metadat i na (vykazovaná) data Osob; interní uživatel musí mít pro svoji práci definován rozsah oprávnění jak pro práci s metapopisem (tedy s „metadaty“), tak při práci s hodnotami údajů vykázaných Osobami (tedy s „daty“), viz kapitola 2.9 Objekt Rozsah oprávnění,
  - **externím uživatelem** je jakýkoli uživatel, který je zástupcem buď Vykazující nebo Zastupující osoby (existuje v Registru osob systému SDAT). Takovému uživateli není za žádných okolností povoleno provádět úpravy metapopisu (tedy „metadat“; proto není potřeba pro externí uživatele jakkoli nastavovat přístupová práva k objektům metapopisu) a Vykazovacích povinností. Externí uživatel je ve většině případů spjat s právě jednou Osobou. Aby Osoba mohla splnit svoji povinnost (dodání dat), musí mít uživatelé této osoby právo vytvářet Hodnoty údajů výkazu, tedy vykazovat data. Ve všech případech platí, že každá Vykazující osoba má právo vytvářet hodnoty údajů sama za sebe a každá Zastupující Osoba má právo vytvářet hodnoty údajů za všechny Vykazující osoby, které zastupuje. Externí uživatele dále dělíme na:
    - **registrované** - registrovaným externím uživatelem je uživatel, který je zástupce Vykazující nebo Zastupující osoby a systém SDAT mu přidělil přihlašovací údaje,
    - **neregistrované** - neregistrovaným externím uživatelem je uživatel, který vstupuje do systému SDAT za účelem vykázání dat, nicméně jeho přístup nepodléhá ověření jeho identity. Tento přístup bude možný pouze pro předkládání předem určených výkazů. Podrobněji je tento způsob přístupu k aplikaci SDAT popsán v dokumentu D – Sběr dat, kapitola 4.3 Kanál Webová aplikace – Neveřejná část s jednorázovým přístupem.

Speciální skupinu uživatelů tvoří **veřejní uživatelé.** To jsou uživatelé, kteří přistupují k veřejné části systému a vyznačují se tím, že pro přístup k metadatům aplikace SDAT (veřejné části) nepotřebují projít autentizačním procesem. Z hlediska dokumentu popisujícího přístupová práva je tato skupina nezajímavá a nebude dále zmiňována. Tito uživatelé nejsou systémem nijak podchyceni a regulováni. Veřejný uživatel metadata ze systému SDAT pouze čte (data metapopisu jsou prezentována na veřejných internetových stránkách ČNB) a nemůže do něj žádná data zapisovat (vykazovat Hodnoty údajů). Aktivita těchto uživatelů je sledována pomocí standardních nástrojů pro monitorování přístupu k webovému serveru ČNB umístěného v síti Internet.

V rámci **zjednodušení správy řízení přístupů** do aplikace SDAT bude každá Osoba vybavena jedním uživatelským místem (zde bude moci být přiřazeno více uživatelů), které poskytne Osobě možnost spravovat veškerá další nastavení, ale pouze v rozsahu oprávnění

předaného z ČNB (tedy v rámci jedné Osoby). V praxi to znamená, že při založení Osoby (ať už autoregistrací nebo klasickým založením) bude vytvořeno „administrátorské“ uživatelské místo (dále nazýváno jako „hlavní uživatelské místo (UM)“ a na toto místo bude zařazen uživatel (v případě autoregistrace jeden; více jich bude možno zařadit v případě klasické registrace prováděné v ČNB). Tito uživatelé získají oprávnění vytvářet další uživatele a uživatelská místa v rámci své Osoby. Tímto způsobem je zajištěno přenesení odpovědnosti za definici oprávnění na stranu Osoby a omezení práce administrátorů systému SDAT.

Některé objekty v oblasti Uživatelé a přístupová práva podléhají sledování časové platnosti. Takové objekty mají v konceptuálním modelu uvedeny atributy „platnost_od“ a „platnost_do“. Na rozdíl od oblasti metapopisu se však jedná o tzv. „reálnou“, nikoli „odloženou“ platnost. To znamená, že – pokud není uvedeno jinak – v této oblasti nelze definovat, že instance objektů platí od jiného, než aktuálního data a času a nelze tak odložit platnost nějaké instance do budoucna.

2.2 Objekt Uživatel

Uživatel je člověk[1], který má oprávnění přistupovat do systému SDAT prostřednictvím aplikačního účtu uživatele a provádět v něm nějaké činnosti (aktivity). V objektovém modelu je aplikační účet uživatele zastoupen objektem Uživatel.

Každý uživatel musí být systémem jednoznačně identifikovatelný, proto platí, že každý uživatel má přidělen právě jeden aplikační účet uživatele. ID aplikačního účtu uživatele musí být jednoznačné a snadno zapamatovatelné. Uživatel jej bude potřebovat zadat při každém pokusu o přihlášení[2]. Z toho důvodu je jako ID aplikačního účtu uživatele požadováno použít e-mailovou adresu uživatele. V případě, že bude třeba toto ID změnit (změna e-mailové adresy může nastat z různých důvodů, například změna příjmení po svatbě), bude tato změna zachycena pomocí vytvoření nové verze (s časovou platností) instance objektu Uživatel.

V rámci objektu Uživatel jsou evidovány tyto atributy (v rámci objektu existují atributy, u kterých změna jejich hodnoty nevyvolává vytvoření nové časové platnosti instance objektu; takové atributy jsou označeny slovním spojením „změna nevytváří novou verzi instance objektu“):

- **systémové ID** – jednoznačné systémové ID, identifikátor je použit pro všechny uživatele bez rozdílu, zda jsou interní nebo externí.
- **platnost_od** – vymezuje časový okamžik, od kterého je uživatel platný (datum a čas vzniku uživatele),
- **platnost_do** – vymezuje časový okamžik, do kterého je uživatel platný. Uplynutím tohoto časového okamžiku pozbývá uživatel platnost (a nelze jej použít pro přihlášení). Je

[1] K systému SDAT mohou přistupovat i stroje (jiné aplikace). Přístup strojů (jiných aplikací) podléhá ověření jejich identity a je realizován skrze aplikační rozhraní (API), více viz dokument E – Výběry dat. Pro tento specifický druh přístupu nejsou používány aplikační účty uživatele.

[2] S výjimkou interních uživatelů, kteří se ověřují pomocí SSO.

dovoleno, aby uživatel toto datum při vytváření uživatele neuvedl; v takovém případě systém automaticky nastaví hodnotu tohoto atributu na maximální datum,

- **typ_uživatele** – možné hodnoty atributu jsou „interní“ nebo „externí“,
- **ID_interní_uživatel** – jednoznačný a významový identifikátor interního uživatele (povinný pouze pro interní uživatele). Tvořen prefixem "U0" a následován čtyřmi čísly, které představují osobní číslo zaměstnance ČNB. Jedná se o username, které je použito při ověření identity (autentifikaci) interního uživatele v rámci SSO. Povinné pro interní uživatele.
- **DB_účet_interní_uživatel** - Název schématu (databázový účet), interního uživatele, který má právo pro přístup k tzv. uživatelským pohledům (Dokument F – Výběry dat, kapitola 5 Uživatelské pohledy), pomocí kterého se uživatel může přihlásit přímo k databázi (mimo systém SDAT) a získávat data z uživatelských pohledů. Uvedením hodnoty do tohoto atributu je uživateli uděleno oprávnění číst data z těchto uživatelských pohledů dle úrovně práv, která jsou nastavena přímo k databázi k danému databázovému účtu. Podrobně viz kapitola 2.10 Vyhodnocení přístupových práv pro uživatelské pohledy.
- **E_mail_interní_uživatel** – e-mailová adresa interního uživatele (externí uživatel má e-mailovou adresu v atributu „ID_externí_uživatel“.
- **ID_externí_uživatel** – jednoznačný (s výjimkou externích neregistrovaných uživatelů, viz níže) a významový identifikátor externího uživatele (povinný pouze pro externí uživatele). Jedná se o e-mailovou adresu externího uživatele, která bude použita pro ověření identity uživatele (autentifikaci). Na tuto e-mailovou adresu bude odesílán e-mail v případě požadavku na obnovu zapomenutého hesla; z tohoto důvodu nebude atribut měnitelný (bezpečnostní riziko); v případě potřeby jej změnit, bude vytvořen nový uživatel. Povinné pro externí uživatele.
- **je_aktivní (změna nevytváří novou verzi instance objektu)** – možné hodnoty atributu jsou „ano“ nebo “ne”. V okamžiku založení nového uživatele (aplikačního účtu uživatele) je standardně nastaveno na „ne“. Toto nastavení trvá do okamžiku, dokud není uživatelem provedena aktivace tohoto účtu pomocí zadání jedinečného aktivačního klíče,
- **aktivační_klíč** - náhodně vygenerovaný jedinečný řetězec znaků, který je systémem vytvořen v okamžiku založení nového uživatele (aplikačního účtu uživatele) a který je následně předán uživateli. Se znalostí ID uživatele a aktivačního klíče je možno provést aktivaci uživatele (nastavit inicializační heslo). Aktivační klíč je používán jednorázově a má smysl pouze v okamžiku, kdy je atribut „je_aktivní“ nastaven na hodnotu „ne“, což nastává jen v okamžiku založení nového uživatele (aplikačního účtu uživatele),
- **platnost_aktivačního_klíče** - datum a čas, do kterého je možno provést aktivaci účtu pomocí zadání jednorázového hesla uvedeného v atributu „aktivační klíč“,
- **je_blokovaný (změna nevytváří novou verzi instance objektu)** - možné hodnoty atributu jsou „ano“ nebo „ne“. Nastavením na „ano“ je uživatel (aplikační účet) dočasně zablokován a není možno se pomocí něj po dobu blokace přihlásit. Blokace uživatelského účtu je provedena administrátorem systému (například v případě podezření na zneužití účtu k neoprávněným aktivitám),
- **je_registrovaný (změna nevytváří novou verzi instance objektu)** - možné hodnoty atributu jsou „ano“ nebo „ne“. V případě potřeby vytvořit neregistrovaného uživatele bude nastaven tento atribut na „ne“; ve všech dalších případech bude na hodnotu „ano“,
- **zamčený_do_kdy (změna nevytváří novou verzi instance objektu)** - Datum/čas, do kdy je aplikační účet uživatele zamčený. Toto datum systém nastaví automaticky v případě, že proběhne N neúspěšných pokusů o přihlášení. Více viz kapitola 4.4 Definice bezpečnostní

politiky. Do doby, než vyprší toto datum/čas, se nelze pomocí tohoto aplikačního účtu uživatele přihlásit. Systém disponuje funkcionalitou, která umožňuje administrátorovi systému provést zrušení hodnoty uložené v tomto atributu a zpřístupnit tak aplikační účet pro přihlášení,

- **jméno a příjmení_uživatele** - identifikace uživatele jeho občanským jménem a příjmením,
- **dočasné_heslo (změna nevytváří novou verzi instance objektu)** – náhodně vygenerovaný jedinečný řetězec znaků, který je systémem vytvořen v okamžiku, kdy uživatel požádá o reset hesla, a který je následně předán uživateli. Se znalostí ID uživatele a dočasného hesla je možno provést změnu hesla daného uživatele, aniž by bylo nutno znát aktuálně platné heslo,
- **platnost_dočasného_hesla (změna nevytváří novou verzi instance objektu)** - datum a čas, do kterého je možno provést změnu hesla pomocí zadání jednorázového hesla uvedeného v atributu „dočasné_heslo“,
- **telefonní_číslo** - telefonní číslo uživatele, na které lze zaslat SMS. Bude využíváno pro rozšířené ověřování identity uživatele (dvoukroková (dvoufaktorová) autentizace). Hodnota toho atributu je měnitelná pouze uživatelem, který je daným účtem spjatý, protože se jedná o údaj, pomocí kterého tento uživatel prokazuje svoji identitu, nemůže ji změnit například administrátor systému (bezpečnostní riziko). V případě, že zároveň pro daného uživatele existuje telefonní číslo v objektu „Kontakt pracovníka“, pak se systém při změně dotáže, zda si uživatel přeje změnit i tento kontakt na stejnou hodnotu. Informace o telefonním čísle v objektu Kontakt pracovníka je pouze informativní a nepoužívá se pro žádnou systémovou aktivitu.

Systém zajišťuje, že hodnota atributu ID externího uživatele (e-mailová adresa) je jedinečné, ale pouze v rámci množiny uživatelů, kteří jsou označeni jako „registrovaní“ (atribut „je_registrovaný = ano“). V rámci množiny instancí objektu Uživatel, které splňují podmínku „je_registrovaný = ne“ není atribut ID externího uživatele definován jako jedinečný, tzn. jednu e-mailovou adresu lze použít jako ID uživatele u více instancí.

Zásadní objekt z hlediska přidělování přístupových práv je **Uživatel na uživatelském místě**, který podchycuje vazbu mezi Uživatelem a Uživatelským místem. Tato vazba je vymezena časovou platností (platnost_od/platnost_do) a určuje, že po dobu platnosti této vazby jsou uživateli přiřazena přístupová práva a rozsah oprávnění, která jsou navázána k uživatelskému místu, ke kterému je uživatel připojen.

Zařazením uživatele na uživatelské místo je možno:

- **trvale** uživateli přiřadit přístupová práva a rozsah oprávnění tak, aby mohl vykonávat svoji práci. V takovém případě se použije vymezení časové platnosti od okamžiku, kdy má změna platit, do „nekonečna“. V případě, že je nutno danému uživateli odebrat přístupová práva a rozsah oprávnění, která získává z titulu zařazení na dané uživatelské místo, pak se provede zadání data/času ukončení platnosti zařazení („nekonečné“ datum se změní na reálné datum, ke kterému má oprávnění zaniknout),
- **dočasně** přiřadit uživateli nějaká přístupová práva a rozsah oprávnění, například z titulu zástupu za jiného uživatele. V takovém případě se instance objektu Uživatel na uživatelském místě vytvoří od reálného data/času s platností do jiného reálného data (kdy má skončit zastupování). Uplynutím data platnosti jsou uživateli automaticky odebrána přístupová práva a rozsah oprávnění, které mu byly přiděleny na omezenou dobu.

2.3 Objekt Certifikáty uživatele

Objekt Certifikáty uživatele představuje úložiště, ve kterém jsou uchovávány všechny informace nutné k ověření, zda poskytnutý certifikát patří uživateli systému SDAT. Systém uchovává informace o všech kvalifikovaných certifikátech (veřejných částech) uživatele v rozsahu:

- vystavitel,
- sériové číslo,

Jeden uživatel smí mít v systému uložen neomezený počet certifikátů. Každý certifikát je vždy navázán k právě jednomu uživateli. Sledování časové platnosti certifikátů na úrovni atributů instance objektu Certifikáty uživatele není třeba, protože certifikát má svoji vlastní platnost a vyhodnocení časové platnosti certifikátu je tak přenecháno aplikační logice, která toto provádí v okamžiku použití certifikátu.

Pro práci s certifikáty bude řešiteli poskytnuto API s názvem „CRT“. Jedná se o interní systém ČNB, který formou svého rozhraní umožní zjistit, zda je předaný certifikát platný nebo ne, případně získat veřejnou část certifikátu za pomocí předané ID autority a Serial Nr. certifikátu.

2.4 Objekt Heslo uživatele

Objekt Heslo uživatele udržuje hesla všech externích registrovaných uživatelů. Interní uživatelé hesla nemají, protože pro ověření jejich identity nejsou potřeba (ověřování identity probíhá pomocí SSO bez nutnosti se prokazovat heslem) a externím neregistrovaným uživatelům není heslo vytvářeno.

Objekt je pomocí kompoziční vazby připojen k objektu Uživatel v kardinalitě 1:N, což znamená, že jeden uživatel smí mít neomezeně hesel (platí, že v jeden časový okamžik je platné právě jedno heslo; neomezeně znamená, že nemusí mít žádné heslo – to je právě případ interních a externích neregistrovaných uživatelů) a že jedno heslo je vždy vázáno k právě jednomu uživateli.

Uchování hesla uživatele v databázi, jeho vznik a exspirace je zcela samostatný problém, který je popsán v části týkající se bezpečnostní politiky (viz kapitola 4 Bezpečnostní politika).

2.5 Objekt Uživatelské místo

Každý uživatel systému musí mít vymezenu množinu akcí (aktivit seskupených do rolí), které smí v systému provádět, tj. přístupová práva. Aby bylo možno tato přístupová práva efektivně spravovat, vzniká institut uživatelské místo. Uživatelské místo představuje základní objekt, ve kterém se spojí uživatel, role (aktivity) a rozsah oprávnění. Tento objekt, pak umožní efektivní správu přístupových práv.

Uživatelské místo tak definuje CO (role/aktivity) je možnost dělat s ČÍM (rozsah oprávnění). Platí, že role a rozsah oprávnění se váží na uživatelské místo, stejně tak, že se na uživatelské místo váží uživatelé. To znamená, že přístupová práva a rozsah oprávnění jsou přidělována uživatelskému místu, nikoli přímo uživateli. Uživatel získá přístupová práva a rozsah

oprávnění tím, že je přiřazen k uživatelskému místu. Nastavení přístupových práv a rozsahu oprávnění je tak provedeno jednou a v případě potřeby jej změnit, se změna provede na jednom místě. Uživatelská místa tak poskytují možnost sdílet stejná přístupová práva více uživatelům najednou.

Objekt Uživatelské místo je spojen asociační vazbou s objektem Osoba. Tato vazba je nepovinná, což znamená, že Uživatelské místo nemusí být asociováno s žádnou Osobou, ale pokud asociováno je, pak je asociováno s právě jednou Osobou. Smyslem této vazby je:

- pro externí uživatelská místa zajistit referenci na Osobu, pro kterou je dané uživatelské místo určeno. Externí uživatelské místo nemůže existovat bez vazby na právě jednu Osobu. Vazba mezi externím uživatelským místem a Osobou zajišťuje, že se uživatelé dané Osoby nedostanou k datům jakékoli jiné Osoby,
- pro interní uživatelské místo umožnit k uživatelskému místu žádnou Osobu nepřipojovat a vytvořit tak uživatelské místo, které přesahuje všechny Osoby v systému. Aparát, jak naopak omezit přístup interních uživatelů pouze k datům určitých Osob, je popsán v dalším textu (viz kapitola 2.9 Objekt Rozsah oprávnění).

Platí tak, že pokud se jedná o interní uživatelské místo (typ_uzivatelskeho_mista = interni), pak je vazba na Osobu nepovinná, naopak v případě, že se jedná o externí uživatelské místo (typ_uzivatelskeho_mista = externi), pak je vazba Uživatele na Uživatelské místo povinná.

Dále pro externí uživatelská místa platí, že k nim smí být přiřazeni jen Uživatelé, kteří jsou připojeni (přes objekt Pracovník Osoby) k téže Osobě, jako je dané uživatelské místo. V rámci objektu Uživatelské místo jsou evidovány tyto atributy:

- **platnost_od** – vymezuje časový okamžik, od kterého je uživatelské místo platné (datum a čas vzniku uživatelského místa),
- **platnost_do** – vymezuje časový okamžik, do kterého je uživatelské místo platné. Uplynutím tohoto časového okamžiku pozbývá uživatelské místo platnost (a nelze jej použít pro přihlášení). Je dovoleno, aby uživatel toto datum při vytváření definice uživatelského místa neuvedl; v takovém případě systém automaticky nastaví hodnotu tohoto atributu na maximální datum,
- **typ_uživatelského_místa** – možné hodnoty atributu jsou „interní“, „externí“. Platí tato základní pravidla:
  - na interní uživatelské místo není možné přiřadit externího uživatele,
  - na externí uživatelské místo není možné zařadit interního uživatele,
- **je_registrované** - možné hodnoty atributu jsou „ano“ nebo „ne“. Hodnotu atributu má smysl vyhodnocovat pouze tehdy, pokud má atribut „typ_uživatelského_místa“ hodnotu „externí“. Nastavením hodnoty atributu na „ne“ znamená, že se jedná o jednorázové uživatelské místo, které je automaticky vytvořeno systémem v případě přístupu neregistrovaného uživatele, a kterému je systémem automaticky nastaven takový rozsah oprávnění, který je třeba k tomu, aby tento neregistrovaný uživatel mohl vykázat hodnoty údajů (zaslat výkaz s daty).
- **je_aktivní** - možné hodnoty atributu jsou „ano“ nebo „ne“. Nastavením na „ne“ je uživatelské místo (třeba i dočasně) zablokováno a uživatelům, k němu přiřazeným, jsou tak odebrána přístupová práva (toto uživatelské místo nelze použít pro přístup do systému SDAT),
- **je_hlavní** - možné hodnoty atributu jsou „ano“ nebo „ne“. Tento atribut má smysl pouze pro externí uživatelská místa (uživatelská místa pro externí neregistrované subjekty nebo

interní uživatelská místa budou mít tento atribut nastaven vždy na „ne“). Další text tak platí pro externí registrovaná uživatelská místa. Hlavní uživatelské místo je vytvořeno při vzniku Osoby (ať už autoregistrací nebo klasickým založením v ČNB) a je(jsou) na něj přiřazen uživatel(é) Osoby, který následně má práva vytvářet další uživatele a uživatelská místa Osoby. Zatímco všichni uživatelé a uživatelská místa v rámci Osoby mohou být libovolně spravována externím uživatelem, hlavní uživatelské místo nikoli (ani jeho atributy, ani nastavení jeho rolí a rozsahu oprávnění). Nastavení hlavního UM smí měnit pouze interní uživatel ČNB disponující patřičným oprávněním. Hlavní UM je právě jedno v rámci Osoby. U hlavního UM není ani možné, aby si uživatel Osoby (externí registrovaný uživatel) měnil jakákoli nastavení s tímto uživatelským místem spjatá. Jakákoli změna musí proběhnout na straně ČNB, kam se obrátí oprávněný uživatel Osoby, pokud chce provést změnu v zařazení uživatelů na hlavní UM,

- **je_vzor** - možné hodnoty atributu jsou „ano“ nebo „ne“. Tento atribut určuje, zda se jedná o reálné uživatelské místo, ke kterému jsou přiřazováni uživatelé (atribut „je_vzor“ pak nabývá hodnoty „ne“) anebo je uživatelské místo definováno jako vzorové (atribut „je_vzor“ pak nabývá hodnoty „ano“). Účelem vzorového uživatelského místa je nadefinovat základní rozsah aktivit na jediném (vzorovém) uživatelském místě, tak aby v budoucnu na základě tohoto nastavení mohlo vzniknout reálné uživatelské místo.
- **Typ_vzoru –** v případě, že se jedná o UM, které je definováno jako vzor, pak tento atribut říká, pro jakou oblast se vzor má použít. V ZD je prozatím definován pouze jeden vzor a to "TYP1". Typ 1 je určen pro vytvoření uživatelského místa v případě autoregistrace, v případě vytvoření hlavního UM pro Osobu, která vzniká v ČNB a v případě neregistrovaného přístupu. V budoucnu tento atribut umožní udržovat v systému různá nastavení vzorových míst, pokud se ukáže, že je to potřeba.

Objekt Uživatelské místo obsahuje nepovinnou rekurzivní asociační vazbu s názvem Rodičovské UM. Účelem této vazby je umožnit definovat závislosti mezi uživatelskými místy. Tato vazba bude použita v případě, že uživatelské místo vznikne klonem ze vzorového uživatelského místa. Klonem ze vzorového uživatelského místa budou vznikat UM v následujících případech:

- založení hlavního uživatelského místa pro nově vzniklou Osobu (externí registrovanou),
- založení uživatelské místa pro autoregistrované Osoby.
- Založení uživatelského místa v případě přístupu neregistrovaného uživatele.

V případě, že vznikne uživatelské místo klonem ze vzorového uživatelského místa, pak je pomocí této rekurzivní vazby uchována informace o tom, z jakého vzorového uživatelského místa nové uživatelské místo vzniklo. Tato vazba bude užitečná pro případ, že by v budoucnu došlo ke změně definice rolí a aktivit na vzorovém uživatelském místě. V takovém případě lze zjistit, jaká všechna uživatelská místa byla ze vzorového uživatelského místa vytvořena a případně k nim zpropagovat změny, které byly provedeny v definici vzorového uživatelského místa. Propagace těchto změn nebude automatická, bude vyžadovat potvrzení uživatele; to znamená, že se uživatel bude moci rozhodnout, zda změny provedené u vzorového uživatelského místa budou propagovány na klonovaná uživatelská místa nebo nikoli.

2.6 Objekt Role

Objekt Role slouží jako pomocný objekt při řízení přístupových práv. Tento objekt seskupuje do logických celků tzv. Aktivity (nejmenší možné jednotky z hlediska definice přístupových práv).

Pokud budeme uvažovat, že instance objektu Aktivita vlastně představují „tlačítka“ (lépe řečeno jakýkoli ovládací prvek uživatelského rozhraní), pomocí kterých uživatel v systému SDAT vykonává svoji práci, pak instance objektu Role umožňuje seskupit Aktivity, které spolu logicky souvisejí, a jedna bez druhé nemá smysl. Typickým příkladem je situace, kdy chceme uživateli přiřadit přístupové právo „Vytvořit číselník“. „Vytvořit číselník“ bude v systému reprezentováno právě jednou Aktivitou. Pokud uživateli dáváme právo vytvořit číselník, bude vhodné mu dát právo „Editovat číselník“ a také „Smazat číselník“. Aby uživatel mohl svoji práci vykonávat opravdu efektivně, bude ještě potřebovat práva „Vytvořit/Editovat/Smazat číselníkovou položku“. To znamená, že pro efektivní práci potřebuje celkem 6 Aktivit. Místo postupného přiřazování každé jedné Aktivity uživatelskému místu založíme roli „Správa číselníku a číselníkových položek“, ke které navážeme 6 výše zmíněných Aktivit. Tuto roli pak přiřadíme uživatelskému místu. Tímto je dosažen kompromis mezi použitelností a spravovatelností systému na jedné straně a potřebou jemného řízení přístupových práv na straně druhé.

Objekt Role obsahuje rekurzivní asociační vazbu s názvem Rodičovská role. Účelem této vazby je umožnění vytvoření nekonečné hierarchie rolí. Smyslem je umožnit vytvořit roli s velkým rozsahem přístupových práv tak, že pod ní seskupíme jednu nebo více existujících Rolí. Pak všechny existující Role budou odkazovat na Rodičovskou roli. Rodičovská role pak bude obsahovat Aktivity všech svých podřízených rolí. Zanoření rolí je nekonečné. Rodičovská role tak nemusí mít přímo přiřazenou žádnou Aktivitu, přesto bude plnit svůj účel, protože „zdědí“ Aktivity od podřízených Rolí.

U každé role bude administrátorem nastaven příznak „typ_přiřazení“, který umožní definovat, zda je role přiřaditelná k uživatelskému místu (případně k jakému typu) či nikoli. Tento příznak bude nabývat hodnot:

- I - Role je přiřaditelná k internímu uživatelskému místu,
- E - Role je přiřaditelná k externímu uživatelskému místu,
- N - Role není přiřaditelná k žádnému typu uživatelského místa.

Účelem tohoto příznaku je omezit seznam rolí, které se budou nabízet uživatelům při nastavování přístupových práv a usnadnit tak orientaci v systému.

Předmětem sledování historie je „obsah“ Role, tedy vazba role na instance objektu Aktivita. Pokud se tento obsah mění (je přiřazována nová aktivita nebo nějaká aktivita je odřazována), je pro danou instanci objektu Role vytvořena nová verze s patřičnou časovou platností.

Role jsou přes objekt **Role na uživatelském místě** přidělovány Uživatelskému místu. Tento zásadní objekt z hlediska přidělování přístupových práv podchycuje vazbu mezi Rolí a Uživatelským místem. Tato vazba je vymezena časovou platností (platnost_od/platnost_do) a určuje, že po dobu platnosti této vazby je uživatelskému místu přiřazena určitá Role, což definuje přístupová práva, která jsou spojena s uživatelským místem.

Jedno uživatelské místo může mít v jeden okamžik přiřazeno N Rolí (neomezeně, tedy i žádnou Roli), v případě, že Uživatelskému místu nebude přiřazena ani jedna Role, nebude

pomocí daného uživatelského místa získána Aktivita (a nebude tak možno získat žádné oprávnění) a uživatelské místo tak nebude mít žádný význam. Jedna Role může být přiřazena více uživatelským místům.

2.7 Objekt Aktivita (Aplikační aktivita)

Objekt Aktivita (pro lepší pochopení je možno též nazývat jako Aplikační aktivita) je objekt, v rámci kterého jsou vytvářeny tzv. aplikační aktivity, neboli z hlediska řízení přístupovým práv nejmenší a pro obsluhu systému dále nedělitelné prvky řízení přístupových práv. Každá aplikační aktivita je tvořena alespoň jednou systémovou aktivitou.

Administrátorovi systému je umožněno pomocí uživatelského rozhraní aplikace jakoukoli aktivitu nastavit jako neaktivní. V případě, že nějaká aktivita bude nastavena jako neaktivní, bude toto zohledněno v procesu vyhodnocování přístupových práv (uživatel nedostane povolení pro provedení systémových aktivit, které jsou k této (aplikační) aktivitě přiřazeny.

Účelem objektu Aktivita je udržovat seznam všech akcí, které jsou implementovány v systému SDAT a které mají být řízeny pomocí přístupových práv, tzn. je třeba při běhu systému a na základě znalosti, o jakého uživatele se jedná, rozhodnout, zda danou akci smí nebo nesmí provést.

Aktivity jsou přes objekt **Aktivita v roli** přidělovány Roli. Tento objekt podchycuje vazbu mezi Aktivitou a Rolí. Tato vazba je vymezena časovou platností (platnost_od/platnost_do) a určuje, že po dobu platnosti této vazby je Roli přiřazena určitá Aktivita.

Jedna role může mít v jeden okamžik přiřazeno N aktivit (neomezeně, tedy i žádnou aktivitu). Role bez aktivity je v pořádku, pokud je daná role tzv. rodičovská, tedy má nějaké podřízené Role s přiřazenými Aktivitami. Pokud by daná role neměla přiřazenu žádnou Aktivitu a ani žádnou podřízenou Roli, která má přiřazenou nějakou Aktivitu, pak nemá daná Role smysl.

2.8 Objekt Systémová aktivita

Účelem objektu Systémová aktivita je udržovat číselník systémových aktivit s jejich přímou vazbou do zdrojového kódu aplikace. Tento objekt je vyloženě systémové povahy a systém SDAT umožňuje pouze zobrazení instancí tohoto objektu, není však umožněna jakákoli manipulace s instancemi tohoto objektu. Manipulace je umožněna pouze pomocí SQL příkazů, které budou (většinou) vykonávány s instalací nové verze systému. Manipulaci s instancemi této třídy provádí výhradně dodavatel systému (anebo ČNB po dohodě s dodavatelem) a dodavatel je odpovědný za obsah tohoto objektu a to i pro předání systému během trvání doby podpory systému.

Dodavatel v tomto objektu vytvoří pro každou jednou atomickou funkcionalitu systému (například: vytvoření číselníku, filtrování v seznamu číselníku, smazání číselníku) jednu instanci a zajistí, že tato instance bude mít jednoznačnou vazbu na zdrojový kód aplikace a na provedení zcela elementární činnosti.

Dále pak dodavatel iniciálně zajistí a během doby trvání podpory průběžně zajišťuje plnění souvisejícího vazebního objektu, který seskupuje systémové aktivity do tzv. aplikačních aktivit (viz N:M asociační vazba objektů Aktivita a Systémová aktivita).

Zároveň objekt Systémová aktivita umožňuje označit ty systémové aktivity, které jsou tzv. defaultní. Takové systémové aktivity budou automaticky přiřazeny všem uživatelům systému SDAT. Jedná se například o takové aktivity jako je „LOGIN“ nebo „PASSWORD_CHANGE“. Definice toho, že je systémová aktivita označena jako defaultní je záležitostí dodavatele systému, stejně tak jako zajištění funkcionality, že se tyto aktivity automaticky přiřadí každému jednomu uživateli. Informace o tom, že systémová aktivita je tzv. defaultní, je prezentovaná administrátorovi systému na patřičném místě uživatelského rozhraní.

Údržba instancí objektu Systémová aktivita a souvisejícího vazebního objektu na objekt Aktivita je v kompetenci dodavatele a nesprávné naplnění těchto objektů je zásadní vadou systému, protože dochází k zásadnímu ovlivnění celého aparátu oprávnění.

2.9 Objekt Rozsah oprávnění

Účelem objektu Rozsah oprávnění je umožnit definovat, k jakým informacím systému SDAT bude mít přístup právě přihlášený uživatel.

Zatímco objekty Role a Aktivita definují tzv. přístupová práva, tedy vymezují rozsah akcí (aktivit), které smí uživatel provádět (definují „CO“ smí uživatel dělat), pak objekt Rozsah oprávnění definuje „S ČÍM“ smí uživatel tyto aktivity dělat.

Standardně jsou přístupová práva řízena pouze přes role a aktivity. Pokud je povolena aktivita „změnit Účet“, pak se má za to, že je uživatel oprávněn změnit JAKÝKOLI existující účet (může editovat jakoukoli instanci objektu Účet; není možno určit, že uživatel může editovat pouze vybrané instance objektu Účet). Pro objekt Účet (ale i další objekty systému) tak není možné definovat Rozsah oprávnění.

Objekt Rozsah oprávnění je využit pro definici rozsahu oprávnění jak pro interní, tak pro externí uživatele.

Pro externí uživatele je potřeba z hlediska rozsahu oprávnění zajistit, aby každá Vykazující osoba měla přístup pouze k datům, která se váží k této Osobě, a každá Zastupující osoba měla přístup pouze k datům, které se váží ke každé Vykazující osobě, kterou daná Zastupující osoba zastupuje. Toho bude docíleno takto – v případě, že je mezi dvěma osobami definován vztah zastupování, například Osoba „X“ zastupuje osobu „Y“, pak je třeba umožnit pracovníkům osoby „X“, aby se dostali k datům osoby „Y“. Nejdříve tak vznikne definice zastupování (osoby se domluví mezi sebou a informují o tom ČNB, ČNB provede samotnou definici zastupování v systému) a následně si administrátor osoby „X“ založí nové uživatelské místo (toto místo bude i nadále napojeno přímou vaznou na Osobu „X“, jinak to ani nebude možné), nicméně systém umožní definovat takový rozsah oprávnění, který navíc umožní definovat práva pro Osobu „Y“. Stále tak platí, že zaměstnanec osoby „X“ může být zařazen pouze na uživatelská místa osoby „X“, nicméně přes definici rozsahu oprávnění se dostane i k datům osoby „Y“.

Systém tak musí umožnit definovat externí uživatelská místa tak, že do rozsahu oprávnění povolí použít pouze Osobu, ke které je uživatelské místo navázané a navíc všechny osoby, pro které je daná Osoba definovaná jako zastupující.

Samotná definice rozsahu oprávnění se vždy skládá z kombinace instancí objektů Osoba a Výkaz. Cílem definice je tedy uvést povolené kombinace osob a výkazů, jejichž data (Hodnoty údajů) smí uživatel vidět (zpracovávat). Definovat přístup k datům pouze pomocí objektů Osoba a Výkaz, tedy taxativně vyjmenovávat kombinace osoby a výkazu, ke kterým je udělován přístup, by však bylo velmi pracné a náročné na údržbu. Proto kromě tohoto taxativního vyjmenovávání kombinací osoba/výkaz (v dalším textu bude nazýváno „statickou definicí rozsahu oprávnění“) je požadováno, aby bylo možno definovat oprávnění přes objekty Typ osoby a Vykazovací rámec. Tento způsob definice (v dalším textu bude nazýván „dynamickou definicí oprávnění“) umožňuje definovat oprávnění tak, že místo toho, aby byly taxativně vyjmenovány osoby/výkazy, je předmětem definice oprávnění skupina, která zastupuje osobu (Typ osoby) nebo výkaz (Vykazovací rámec). Tím, že je místo taxativně vyjmenované kombinace osoba/výkaz uvedena kombinace typ skupiny/vykazovací rámec, je dosaženo jisté úrovně „bezúdržbovosti“, tj. v případě, že je do konkrétního typu osoby zařazena nová osoba (například vznikne-li nová banka), není třeba definici rozsahu oprávnění měnit. Zařazením osoby do daného typu je pak zajištěno, že pokud uživatel má oprávnění na daný typ, kam byla nová osoba zařazena, má oprávnění i na novou osobu. Stejný princip je pak aplikován pro výkaz; vznikne-li nový výkaz, který je zařazen do vykazovacího rámce, ke kterému má uživatel definováno oprávnění, získá uživatel automaticky oprávnění k datům nově vzniklého výkazu. Zařazováním osob do typu osoby resp. výkazu do vykazovacího rámce tak přímo ovlivňuje rozsah oprávnění (zvětšuje celkovou množinu viditelných dat). I když je rozsah oprávnění definován dynamicky, tak je vždy zpětně převoditelný na taxativní výčet kombinací osoba/výkaz. Toto převedení je potřeba v okamžiku, kdy systém má vyhodnotit definovaný rozsah oprávnění a také v okamžiku, kdy si uživatel chce prohlédnout, data jakých konkrétních osob a výkazů má právo vidět.

Vzhledem k tomu, že je potřeba řídit i přístupová práva a rozsah oprávnění interních uživatelů pro samotné projektování (jde o možnost určit, kteří uživatelé smí upravovat (projektovat) které výkazy), je třeba rozlišit definici oprávnění i z tohoto hlediska. V dalším textu bude odlišena definice oprávnění pro „data“ a „metadata“. Zatímco definice oprávnění pro přístup k datům se skládá z definice kombinace osoba/výkaz, pro definici rozsahu oprávnění k metadatům je třeba definovat pouze to, ke kterým výkazům má mít uživatel (pro projektování) přístup; objekt Osoba není z hlediska definice rozsahu oprávnění pro metadata relevantní. I v případě definice oprávnění pro metadata platí, že je možno definovat jak staticky (taxativním vyjmenováním výkazů), tak dynamicky (určením vykazovacího rámce).

Definici rozsahu oprávnění tak lze provést takto:

- metadata
    - staticky – vyjmenováním konkrétního výkazu (konkrétních výkazů),
    - dynamicky – vyjmenováním konkrétního vykazovacího rámce (vykazovacích rámců).
- data
    - staticky – vyjmenováním konkrétních kombinací osoba/výkaz,
    - dynamicky – vyjmenováním konkrétních kombinací typ osoby/vykazovací rámec,
    - kombinovaně
        - vyjmenováním konkrétních kombinací typ osoby/výkaz,

- vyjmenováním konkrétních kombinací osoba/vykazovací rámec.

V rámci výše uvedených způsobů definice oprávnění je třeba zohlednit následující:

- v případě dynamické definice rozsahu oprávnění (ať už pro data nebo metadata) je třeba zajistit způsob definice výjimek. Výjimky umožňují vyřešit situaci, kdy je požadováno, aby uživatel získal přístup k celé skupině (typu osoby nebo vykazovacímu rámci) s výjimkou několika málo konkrétních osob/výkazů. V takovém případě je uživateli přiděleno oprávnění na danou skupinu a následně se z této skupiny vyčlení určité konkrétní prvky. Tímto bude dosaženo, že uživatel získá oprávnění ke všem současným (a budoucím) prvkům množiny a naopak nebude oprávněn přistupovat k prvkům množiny, které byly definovány výjimkou. Takový způsob definice oprávnění je nazýván „negativní výjimka z dynamické definice oprávnění“,
- v případě statické definice rozsahu oprávnění (ať už pro data nebo metadata) je třeba, kromě taxativního výčtu osob/výkazů, umožnit zápis typu „všechny osoby“ nebo „všechny výkazy“. Tento způsob definice statického oprávnění je nazýván „statická definice oprávnění pomocí hvězdičkové konvence“.
- v případě definice oprávnění pro data je třeba řídit přístup k tzv. citlivým výkazům[3]. Z tohoto hlediska platí pravidlo, že k citlivým výkazům lze udělit oprávnění pouze těmito způsoby:
  - definice osoby je provedena staticky, definice výkazu je provedena staticky,
  - definice osoby je provedena dynamicky, definice výkazu je provedena staticky.

  K citlivým výkazům tak není možno definovat oprávnění v případě, že je použit dynamický způsob definice oprávnění na straně výkazů.

Podrobně jsou všechny přípustné kombinace a způsoby přidělování přístupových práv popsány v kapitolách 2.9.7 Rozsah oprávnění – možnosti definice pro oblast „metadata“ a 2.9.8 Rozsah oprávnění – možnosti definice pro oblast „data“.

2.9.1 Klíčové atributy objektu Rozsah oprávnění

Objekt Rozsah oprávnění je základním objektem pro definici rozsahu oprávnění. V rámci tohoto objektu jsou definovány tyto klíčové atributy:

- **platnost_od** – vymezuje časový okamžik, od kterého definice rozsahu oprávnění nabývá platnosti,
- **platnost_do** – vymezuje časový okamžik, do kterého je definice rozsahu oprávnění platná. Uplynutím tohoto časového okamžiku pozbývá rozsah oprávnění platnosti. Je dovoleno, aby uživatel toto datum při vytváření definice rozsahu oprávnění neuvedl; v takovém případě systém automaticky nastaví hodnotu tohoto atributu na maximální datum,

[3] Za citlivý výkaz je považován takový výkaz, který obsahuje alespoň jednu datovou oblast, která obsahuje alespoň jeden citlivý údaj.

- **typ_oprávnění** – definice toho, pro jakou oblast je rozsah oprávnění definován. Atribut může nabývat pouze dvou hodnot a to buď „data“ (pak se definice oprávnění týká Hodnot údajů) anebo „metadata“ (pak se definice oprávnění týká projektování výkazů),
- **povolit_citlivé_údaje** – atribut typu „boolean“, může nabývat pouze hodnot „ano“ a „ne“, přičemž defaultní hodnota je „ne“. Tento atribut má smysl definovat a vyhodnocovat pouze tehdy, pokud je hodnota atributu „typ_oprávnění = data“. V případě, že je hodnota atributu „povolit_citlivé_údaje = ano“, pak to je systémem interpretováno tak, že pokud se v rámci definice daného rozsahu oprávnění vyskytuje citlivý výkaz, pak je uživateli uděleno oprávnění pracovat i s datovými oblastmi/údaji, které jsou v rámci daného výkazu označeny jako citlivé. V opačném případě (hodnota atributu „povolit_citlivé_údaje = ne“) to znamená, že pokud se v rámci definice daného rozsahu oprávnění vyskytuje citlivý výkaz, pak není uživateli uděleno oprávnění pracovat i s datovými oblastmi/údaji, které jsou v rámci daného výkazu označeny jako citlivé, pak uživatel smí s daty takového výkazu pracovat, ale nebude mít přístup k citlivým údajům (o této skutečnosti ho systém zřetelně informuje). Systém zajišťuje, že hodnota atributu „povolit_citlivé_údaje“ bude vždy „ne“ v případě, že uživatel bude definovat rozsah oprávnění pro výkazy dynamicky,
- **způsob_definice_osob** – určení toho, jakým způsobem bude definován rozsah oprávnění z hlediska přístupů k instancím objektu Osoba. Tento atribut má smysl definovat a vyhodnocovat pouze tehdy, pokud je hodnota atributu „typ_oprávnění = data“. V případě, že je hodnota atributu „typ_oprávnění = metadata“, smí být hodnota tohoto atributu NULL. V případě, že je hodnota atributu „typ_oprávnění = data“, pak atribut „způsob_definice_osob“ smí nabývat buď hodnoty „staticky“ anebo „dynamicky“. Tímto je určeno, jak následně proběhne definice rozsahu oprávnění pro přístup k instancím objektu Osoba:
  - v případě, že je hodnota atributu „způsob_definice_osob = staticky“, pak je rozsah oprávnění určen na konkrétní instance objektu Osoba, případně není určen vůbec (tzv. hvězdičková konvence),
  - v případě, že je hodnota atributu „způsob_definice_osob = dynamicky“, pak je rozsah oprávnění určen na konkrétní instance objektu Typ osoby.
- **způsob_definice_výkazů** – určení toho, jakým způsobem bude definován rozsah oprávnění z hlediska přístupů k instancím objektu Výkaz. Atribut smí nabývat buď hodnoty „staticky“ anebo „dynamicky“. Tímto je určeno, jak následně proběhne definice rozsahu oprávnění pro přístup k instancím objektu Výkaz:
  - v případě, že je hodnota atributu „způsob_definice_výkazů = staticky“, pak je rozsah oprávnění určen na konkrétní instance objektu Výkaz, případně není určen vůbec (tzv. hvězdičková konvence, to znamená právo přístupu ke všem výkazům),
  - v případě, že je hodnota atributu „způsob_definice_výkazů = dynamicky“, pak je rozsah oprávnění určen na konkrétní instance objektu Vykazovací rámec.
- **ignorovat_datumovou_platnost (boolean, default „false“)** – v případě, že je rozsah oprávnění definován (alespoň z části) dynamicky, dochází během procesu vyhodnocení přístupových práv k překladu dynamické definice na statickou definici. V rámci tohoto procesu je standardně zkoumaná datumová platnost (datumová platnost zařazení osoby do typu osoby a datumová platnost zařazení výkazu do vykazovacího rámce). Pomocí nastavení tohoto příznaku na hodnotu „true“ lze toto pravidlo (vyhodnocování datumové platnosti) potlačit; v takovém případě budou do oprávnění zahrnuty i ty výkazy/osoby, které jsou sice přiřazeny k vykazovacímu rámci/osobě, ale toto přiřazení není k okamžiku vyhodnocení přístupových práv již platné (rozhodný okamžik vyhodnocování

přístupových práv leží mimo interval zařazení výkazu do vykazovacího rámce a osoby do typu osoby).

2.9.2 Vazba objektů Rozsah oprávnění a Osoba

Vazba objektu Rozsah oprávnění a Osoba je definována jako N:M, což umožňuje v rámci jedné instance objektu Rozsah oprávnění přivázat žádnou, jednu nebo více instancí objektu Osoba a stejná instance objektu Osoba může být přivázána k více různým instancím objektu Rozsah oprávnění. Není sledována časová platnost vazby, ta se váže k určité instanci objektu Rozsah oprávnění a je tak definovaná pro jeho časovou platnost. Vazba mezi instancemi obou objektů může vzniknout pouze tehdy, pokud uživatel definuje rozsah oprávnění staticky a pro data („typ_oprávnění = data“ a „způsob_definice_osob = staticky“ a to takto:

- v případě, že je požadováno, aby byl rozsah oprávnění definován na všechny instance objektu Osoba, které aktuálně v systému existují, a zároveň na všechny instance objektu Osoba, které vzniknou v budoucnu, pak nevznikne žádná vazba mezi instancí objektu Rozsah oprávnění a jakoukoli instancí objektu Osoba (systém tento zápis při vyhodnocování přístupových práv vyhodnotí jako „všechny osoby“, jedná se tedy o hvězdičkovou konvenci),
- v případě, že je požadováno, aby byl rozsah oprávnění definován na konkrétní (v daný okamžik existující) instance objektu Osoba, jsou tyto instance určeny uživatelem (multi-výběrem ze seznamu osob) a pro každou takovou vybranou Osobu vznikne vazba na právě vytvářenou instanci objektu Rozsah oprávnění.

2.9.3 Vazba objektů Rozsah oprávnění a Typ osoby

Vazba objektu Rozsah oprávnění a Typ osoby je definována jako N:M, což umožňuje v rámci jedné instance objektu Rozsah oprávnění přivázat žádnou, jednu nebo více instancí objektu Typ osoby a stejná instance objektu Typ osoby může být přivázána k více různým instancím objektu Rozsah oprávnění. Není sledována časová platnost vazby, ta se váže k určité instanci objektu Rozsah oprávnění a je tak definovaná pro jeho časovou platnost. Vazba mezi instancemi obou objektů může vzniknout pouze tehdy, pokud uživatel definuje rozsah oprávnění dynamicky a pro data („typ_oprávnění = data“ a „způsob_definice_osob = dynamicky“), a to tak, že jsou instance objektu Typ osoby určeny uživatelem (multi-výběrem ze seznamu Typu osob) a pro každý takový vybraný Typ osoby vznikne vazba na právě vytvářenou instanci objektu Rozsah oprávnění.

2.9.4 Vazba objektů Rozsah oprávnění a Výkaz

Vazba objektu Rozsah oprávnění a Výkaz je definována jako N:M, což umožňuje v rámci jedné instance objektu Rozsah oprávnění přivázat žádnou, jednu nebo více instancí objektu Výkaz a stejná instance objektu Výkaz může být přivázána k více různým instancím objektu Rozsah oprávnění. Není sledována časová platnost vazby, ta se váže k určité instanci objektu Rozsah oprávnění a je tak definovaná pro jeho časovou platnost. Vazba mezi instancemi obou objektů může vzniknout pouze tehdy, pokud uživatel definuje rozsah oprávnění staticky

(„způsob_definice_výkazů = staticky“) bez ohledu na to, zda se jedná o definici přístupů k datům nebo metadatům, a to takto:

- v případě, že je požadováno, aby byl rozsah oprávnění definován na všechny instance objektu Výkaz, které aktuálně v systému existují, a zároveň na všechny instance objektu Výkaz, které vzniknou v budoucnu, pak nevznikne žádná vazba mezi instancí objektu Rozsah oprávnění a jakoukoli instancí objektu Výkaz (systém tento zápis při vyhodnocování přístupových práv vyhodnotí jako „všechny výkazy“, jedná se tedy o hvězdičkovou konvenci),
- v případě, že je požadováno, aby byl rozsah oprávnění definován na konkrétní (v daný okamžik existující) instance objektu Výkaz, jsou tyto instance určeny uživatelem (multi-výběrem ze seznamu výkazů) a pro každou takovou vybranou Osobu vznikne vazba na právě vytvářenou instanci objektu Rozsah oprávnění.

2.9.5 Vazba objektů Rozsah oprávnění a Vykazovací rámec

Vazba objektu Rozsah oprávnění a Vykazovací rámec je definována jako N:M, což umožňuje v rámci jedné instance objektu Rozsah oprávnění přivázat žádnou, jednu nebo více instancí objektu Vykazovací rámec a stejná instance objektu Vykazovací rámec může být přivázána k více různým instancím objektu Rozsah oprávnění. Není sledována časová platnost vazby, ta se váže k určité instanci objektu Rozsah oprávnění a je tak definovaná pro jeho časovou platnost. Vazba mezi instancemi obou objektů musí vzniknout tehdy, pokud uživatel definuje rozsah oprávnění dynamicky („způsob_definice_výkazů = dynamicky“) bez ohledu na to, zda se jedná o definici přístupů k datům nebo metadatům, a to tak, že jsou instance objektu Vykazovací rámec určeny uživatelem (multi-výběrem ze seznamu vykazovacích rámců) a pro každý takový vybraný Vykazovací rámec vznikne vazba na právě vytvářenou instanci objektu Rozsah oprávnění.

2.9.6 Definice výjimek z dynamicky definovaného rozsahu oprávnění

Systém umožňuje definovat výjimky z dynamicky definovaného rozsahu oprávnění. Vzhledem k tomu, že se definice rozsahu oprávnění může skládat ze dvou částí, lze i výjimky definovat ve dvou částech.

Výjimka může být definovaná:

- na Osobu (výjimka vůči dynamické definici přes objekt „Typ osoby“),
- na Výkaz (výjimka vůči dynamické definici přes objekt „Vykazovací rámec“),
- na osobu i na výkaz (v případě dynamické definice přes objekty „Typ osoby“ a „Vykazovací rámec“

Smyslem výjimky z rozsahu oprávnění je snazší správa rozsahu oprávnění, kdy je žádoucí přidělit oprávnění k nějaké skupině (ať už ke skupině Osob, které jsou v systému zastoupeny objektem Typ osoby, tak ke skupině Výkazů, které jsou v systému zastoupeny objektem Vykazovací rámec) s tím, že ne ke všem prvkům dané skupiny má uživatel získat přístup.

Výjimka se tak použije tehdy, pokud má být uživateli přiděleno oprávnění na všechny banky (předpokládejme, že v systému bude existovat „typ osoby = banky“) s výjimkou banky „Banka ABC“. Definici tohoto typu rozsahu oprávnění by bylo možno zapsat i staticky

(uživatel by vybral všechny banky, kromě banky „Banka ABC“), nicméně se vznikem nové banky by bylo nutné takto definované oprávnění upravit. Místo toho vznikne dynamická definice rozsahu oprávnění na typ osoby = banka a zároveň bude definovaná negativní výjimka, která určí, že do rozsahu oprávnění nemá být zahrnuta banka „Banka ABC“. Ekvivalentně lze popsat situaci i na straně definice přístupů k Výkazům.

Pro definici výjimek je v objektovém modelu určen objekt „Definice výjimky z rozsahu oprávnění“. Tento objekt je připojen kompoziční vazbou s kardinalitou 1:N k objektu Rozsah oprávnění, což znamená, že jedna instance objektu Rozsah oprávnění může mít žádnou, jednu nebo více definovaných výjimek, ale daná výjimka se vždy vztahuje k právě jedné instanci objektu Rozsah oprávnění, a se zánikem instance objektu Rozsah oprávnění zanikají i všechny podřízené instance objektu „Definice výjimky z rozsahu oprávnění“.

Výjimka z rozsahu oprávnění může být definována pouze tehdy, pokud alespoň jedna strana definice rozsahu oprávnění je definována jako dynamická (viz atributy objektu Rozsah oprávnění „způsob_definice_osob“ a „způsob_definice_výkazů“). V případě, že obě strany definice rozsahu oprávnění jsou definovány staticky, není možné výjimku definovat (systém v takovém případě zajišťuje, že nevznikne žádná související instance objektu Definice výjimky z rozsahu oprávnění).

Objekt Definice výjimky z rozsahu oprávnění je dále navázán na objekty Osoba a Výkaz. Oba objekty jsou připojeny s kardinalitou 0..1, což znamená, že napojení instancí obou objektů na instanci objektu „Definice výjimky z rozsahu oprávnění“ je nepovinné. Díky takto postaveným vazbám je možno zajistit, že výjimka bude definována pouze na jedné straně (buď na straně Výkazu anebo na straně Osoby). Systém musí zajistit, aby:

- nebylo možno definovat výjimku, která nebude mít definovanou ani jednu stranu; to znamená, že nesmí vzniknout instance objektu Definice výjimky z rozsahu oprávnění, která bude mít žádnou Osobu a žádný Výkaz,
- byla vždy definována ta strana výjimky, pro kterou je rozsah oprávnění definován dynamicky. To znamená, že:
    - pokud je nadřízená instance objektu Rozsah oprávnění definována způsobem „způsob_definice_osob = dynamicky“ a způsob_definice_výkazů = staticky“, může být výjimka definována pouze výčtem Osob,
    - pokud je nadřízená instance objektu Rozsah oprávnění definována způsobem „způsob_definice_osob = staticky“ a způsob_definice_výkazů = dynamicky“, může být výjimka definována pouze výčtem výkazů,
    - pokud je nadřízená instance objektu Rozsah oprávnění definována způsobem „způsob_definice_osob = dynamicky“ a způsob_definice_výkazů = dynamicky“, může být výjimka definována jak výčtem Osob, tak Výkazů,
- nebylo možno definovat výjimku na straně Osob v případě, že nadřízená instance objektu Rozsah oprávnění je definována pro metadata.

2.9.7 Rozsah oprávnění – možnosti definice pro oblast „metadata“

Tabulka č. 1 popisuje všechny možné způsoby definice oprávnění pro řízení přístupu k metadatům (projektování výkazů)

ID	Způsob definice Výkazů	Vazba na objekt Vykazovací rámec	Vazba na objekt Výkaz	Výjimky
A	STATICKY	**NULL**	**A***	Výjimky nelze definovat.
B	DYNAMICKY	**A**	**NULL**	Výjimky lze definovat pro Výkazy.

Tabulka 1- Možné způsoby definice rozsahu oprávnění pro metadata

Legenda:
NULL – vazba na instanci daného objektu neexistuje.
A – vazba na instanci daného objektu existuje.
A* - vazba na instanci daného objektu existuje; pokud neexistuje, systém provádí substituci za „*“ (všechny instance daného objektu).

2.9.8 Rozsah oprávnění – možnosti definice pro oblast „data“

Tabulka č. 2 popisuje všechny možné způsoby definice oprávnění pro řízení přístupu k datům (Hodnoty údajů).

ID	Způsob definice Osob	Způsob definice Výkazů	Vazba na objekt Typ osoby	Vazba na objekt Osoba	Vazba na objekt Vykazovací rámec	Vazba na objekt Výkaz	Výjimky
A	STATICKY	STATICKY	**NULL**	**A***	**NULL**	**A***	Výjimky nelze definovat.
B	STATICKY	DYNAMICKY	**NULL**	**A***	**A**	**NULL**	Výjimky lze definovat jen pro Výkazy.
C	DYNAMICKY	STATICKY	**A**	**NULL**	**NULL**	**A***	Výjimky lze definovat jen pro Osoby.
D	DYNAMICKY	DYNAMICKY	**A**	**NULL**	**A**	**NULL**	Výjimky lze definovat jak pro Osoby, tak pro Výkazy.

Tabulka 2 - Možné způsoby definice rozsahu oprávnění pro data

Legenda:
NULL – vazba na instanci daného objektu neexistuje.
A – vazba na instanci daného objektu existuje.
A* - vazba na instanci daného objektu existuje; pokud neexistuje, systém provádí substituci za „*“ (všechny instance daného objektu).

2.9.9 Definice rozsahu oprávnění pro „metadata“

V případě, že je instance objektu Rozsah oprávnění definována pro „metadata“, pak se má za to, že se jedná o definici rozsahu oprávnění pouze pro oblast projektování výkazů. V takovém případě je nutno definovat rozsah oprávnění pouze tak, že **je třeba určit rozsah Výkazů, na které má uživatel oprávnění.** Z hlediska objektového modelu vzniká instance objektu Rozsah oprávnění, kde atribut „typ_oprávnění“ nabývá hodnoty „metadata“.

V případě, že se jedná o typ oprávnění pro metadata, pak nemá smysl uvažovat o přidělování oprávnění pro citlivá data. Atribut „povolit_citlivé_údaje“ tak bude nabývat hodnoty „ne“ (jedná se o atribut typu boolean, takže nějaká hodnota daného atributu musí existovat; z hlediska vyhodnocování oprávnění typu metadata nebude tato hodnota nijak zohledněna).

Dále platí, že pro metadata není možné definovat stranu oprávnění Osoba (konkrétní Osoby ani skupinu Osob přes objekt Typ osoby). Proto pro danou instanci objektu Rozsah oprávnění tak dále platí, že hodnota atributu „způsob_definice_osob“ je NULL.

Uživatel při definici rozsahu oprávnění pro metadata musí určit, zda definici provede staticky nebo dynamicky. Tato skutečnost je podchycena atributem „způsob_definice_výkazů“. Na základě toho systém buď nabídne seznam výkazů (statická definice; včetně položky „všechny výkazy“ (hvězdičková konvence)), anebo seznam vykazovacích rámců (dynamická definice). Uživatel vybere buď jeden nebo více výkazů (nebo položku „všechny výkazy“) v případě statické definice oprávnění nebo jeden nebo více vykazovacích rámců, v případě dynamické definice rozsahu oprávnění.

Mohou tak vzniknout následující situace:

- instance objektu Rozsah oprávnění je typu metadata a je použit statický způsob definice rozsahu oprávnění:
  - neexistuje-li žádná vazba dané instance na objekt Výkaz, znamená to, že uživatel udělil právo na všechny výkazy,
  - existuje-li alespoň jedna vazba dané instance na objekt Výkaz, znamená to, že uživatel udělil právo na daný Výkaz (na dané Výkazy, pokud existuje vazeb více),
- instance objektu Rozsah oprávnění je typu metadata a je použit dynamický způsob definice rozsahu oprávnění – pak musí existovat alespoň jedna vazba mezi objekty Rozsah oprávnění a Vykazovací rámec. Přiřazení instance(í) objektu Vykazovací rámec se použije pro definici rozsahu oprávnění pouze v případě, že je žádoucí, aby měl uživatel rozsah oprávnění definován dynamicky v rozsahu vykazovacího rámce (vykazovacích rámců), který je mu přidělen. Tento způsob definice rozsahu oprávnění nebude použit v případě, že je potřeba nastavit rozsah oprávnění na takovou množinu výkazů, která neodpovídá žádnému konkrétnímu vykazovacímu rámci, pak žádná instance objektu Vykazovací rámec nebude k instanci objektu Rozsah oprávnění přiřazena a definice rozsahu oprávnění bude provedena staticky přes objekt Výkaz. Více o výjimkách z dynamicky definovaného rozsahu oprávnění viz kapitola 2.9.6 Definice výjimek z dynamicky definovaného rozsahu oprávnění.

Přiřazením každé jedné instance objektu Vykazovací rámec k instanci objektu Rozsah oprávnění se má za to, že je uděleno právo na projektování všech výkazů, které jsou (v okamžiku potřeby vyhodnotit rozsah oprávnění) zařazeny do daného vykazovacího rámce. S ohledem na fakt, že výkazy jsou do vykazovacího rámce zařazovány s časovou platností (atributy „zařazen_dne“ a „vyřazen_dne“ v objektu „Výkaz ve Vykazovacím rámci“, znamená to, že musí být tato časová platnost zařazení při vyhodnocování rozsahu oprávnění zohledněna. To znamená, že pokud systém vyhodnocuje rozsah oprávnění pro uživatele X, přiřazeného na uživatelském místě Y k časovému okamžiku například 15. 1. 2015 10:30:00, pak se nejdříve zjistí, jaké instance objektu Rozsah oprávnění jsou platné k danému časovému okamžiku. Pokud je například výsledkem tohoto zjištění rozsah oprávnění na vykazovací rámec VR1, pak se dále zjišťuje, jaké výkazy jsou zařazeny do vykazovacího rámce VR1 v čase 15. 1. 2015 10:30:00. V případě, že je sice výkaz V1 zařazen do vykazovacího rámce VR1, ale platnost tohoto zařazení byla ukončena k 14. 1. 2015 ve 23:59:59, pak to znamená, že oprávnění pro projektování na výkaz V1 nebude uděleno (časová platnost zařazení Výkazu V1 do vykazovacího rámce, ke kterému má uživatel udělen přístup, již vypršela).

2.9.10 Definice rozsahu oprávnění pro „data“

V případě, že je instance objektu Rozsah oprávnění definována typem „DATA“, pak se má za to, že se jedná o definici rozsahu oprávnění pouze pro oblast čtení vykázaných Hodnot údajů Osobami.

V takovém případě je nutno definovat rozsah oprávnění takto:

- **je třeba určit rozsah Výkazů, na které má uživatel oprávnění.** To se zajistí připojením jedné nebo více instancí objektu Výkaz nebo Vykazovací rámec k objektu Rozsah oprávnění podle toho, zda se jedná o statický nebo dynamický způsob definice rozsahu oprávnění. Jedná se o stejný způsob definice rozsahu oprávnění jako v případě typu rozsahu oprávnění „metadata“, proto zde nebude již znovu popisován, více viz kapitola 2.9.9. Definice rozsahu oprávnění pro „metadata“. Výjimku tvoří situace, kdy zařazení do Vykazovacího rámce není k okamžiku vyhodnocování oprávnění časově platné, ale pro daný Vykazovací rámec je nastaven atribut „ponechat_přístup_k_datům“ na hodnotu „ano“. Popis definice výjimek z dynamického rozsahu oprávnění pro Výkazy je popsán v kapitole 2.9.6 Definice výjimek z dynamicky definovaného rozsahu oprávnění.
- **je třeba určit rozsah Osob, na které má uživatel oprávnění.** To se zajistí připojením jedné nebo více instancí objektu Osoba nebo Typ osoby k objektu Rozsah oprávnění, podle toho, zda se jedná o statický nebo dynamický způsob definice rozsahu oprávnění. Přiřazení instance(í) objektu Typ osoby se použije pro definici rozsahu oprávnění pouze v případě, že je žádoucí, aby měl uživatel rozsah oprávnění definován dynamicky v rozsahu určité skupiny subjektů (Typů osoby), která je mu přidělena. Tento způsob definice rozsahu oprávnění nebude použit v případě, že je potřeba nastavit rozsah oprávnění na takovou množinu Osob, která neodpovídá žádné existující skupině a definice rozsahu oprávnění bude provedeno staticky. Více viz kapitola 2.9.6 Definice výjimek z dynamicky definovaného rozsahu oprávnění.

Rozsah oprávnění pro „data“ je tak kartézským součinem všech prvků množiny Osoba se všemi prvky množiny Výkaz. V případě, že je definice rozsahu oprávnění provedena staticky, jsou kombinace osoba/výkaz reprezentovány instancemi patřičných objektů, v případě

dynamického způsobu definice je nutné tyto kombinace „vypočítat“ na základě toho, jaké Osoby jsou zařazeny do přiřazeného Typu Osoby a jaké Výkazy jsou zařazeny do daného Vykazovacího rámce; navíc je v případě dynamického způsobu definice rozsahu oprávnění potřeba zohlednit výjimky. Pokud bude rozsah oprávnění typu „data“ definován takto:

- Typ Osoby: Banky,
- Vykazovací rámec: COREP, FINREP,

pak to znamená (za předpokladu neexistence výjimek z této definice), že uživatel smí vidět data vykázaná všemi Osobami, které jsou zařazeny k okamžiku vyhodnocování rozsahu oprávnění v typu osoby „Banky“ (tím je dána množina Osob) za všechny Výkazy, které jsou zařazeny buď ve vykazovacím rámci COREP nebo FINREP (tím je dána množina Výkazů).

Jak Osoba do Typu osoby, tak Výkaz do Vykazovacího rámce je zařazen s vymezením časové platnosti. Při vyhodnocování rozsahu oprávnění je tak třeba tuto časovou platnost zohlednit. Způsob zohlednění časové platnosti je shodný se způsobem popsaným u vyhodnocování rozsahu oprávnění pro metadata, viz kapitola 2.9.9 Definice rozsahu oprávnění pro „metadata“.

2.9.11 Definice rozsahu oprávnění pro citlivé výkazy

Za citlivý výkaz se považuje výkaz, který obsahuje alespoň jednu datovou oblast, která obsahuje alespoň jeden citlivý údaj (každý Údaj nese o sobě informaci, zda je citlivý nebo ne, viz dokument B – Metapopis).

Přístup k citlivým datům je možno zajistit pouze tehdy, pokud je definice rozsahu oprávnění provedena staticky na straně Výkazu. Na straně Osoby je možno použít statickou i dynamickou definici rozsahu oprávnění. O tom, zda je nebo není povolen přístup k citlivým údajům, rozhoduje atribut „povolit_citlivé_údaje“, který je součástí objektu Rozsah oprávnění. Podrobně je způsob práce s tímto atributem popsán v kapitole 2.9.1 Klíčové atributy objektu Rozsah oprávnění.

V rámci definice dynamického rozsahu oprávnění pro typ oprávnění „data“ přes objekt Vykazovací rámec se má za to, že je-li tímto dynamickým rozsahem oprávnění uděleno právo přístupu k hodnotám údajů v rámci Výkazu, který obsahuje citlivé údaje, tak se toto oprávnění vztahuje pouze na ty datové oblasti, které neobsahují žádný citlivý údaj. To znamená, že pokud je uživateli přiřazen rozsah oprávnění na vykazovací rámec VR1, který obsahuje 10 výkazů, a dva z těchto deseti výkazů obsahují aspoň jeden citlivý údaj, pak je oprávnění uděleno sice na všech 10 výkazů, ale u dvou výkazů, které obsahují citlivá data, uživatel získá přístup pouze k hodnotám údajů, které jsou zařazeny do datových oblastí, které neobsahují ani jeden citlivý údaj (hodnoty údajů z datové oblasti s citlivými údaji nejsou dostupné, a to ani ty hodnoty, které samy o sobě nejsou označeny jako citlivé).

Aby bylo možno získat oprávnění i na datové oblasti, které obsahují citlivá data, je třeba nadefinovat samostatný rozsah oprávnění (vytvořit novou instanci objektu Rozsah oprávnění), v rámci které bude vymezen okruh osob a výkazů, pro které bude umožněn přístup k citlivým údajům s tím, že strana Výkazů musí být definována staticky.

Přístup k citlivým datovým oblastem se definuje vždy přes nadřízený objekt Výkaz. Tímto přístupem lze buď udělit oprávnění na všechny citlivé datové oblasti v rámci

výkazu, nebo na žádnou; řízení přístupových práv na citlivé údaje tedy neumožňuje taxativně vyjmenovávat, ke kterým citlivým datovým oblastem uživatel získá oprávnění a ke kterým nikoli).

V rámci definice rozsahu oprávnění k citlivým výkazům systém nabídne uživateli seznam všech Výkazů, které obsahují alespoň jednu datovou oblast, která obsahuje alespoň jeden citlivý údaj. Uživatel vybere ty citlivé výkazy, ke kterým chce povolit uživatelům přiřazeným k uživatelskému místu přístup. Následně je nutno definovat rozsah Osob, pro které toto oprávnění platí (rozsah osob lze definovat staticky i dynamicky), a následně vznikne rozsah oprávnění na citlivé datové oblasti.

2.9.12 Vyhodnocení definice rozsahu oprávnění

Následující text popisuje algoritmus, jakým bude rozsah oprávnění - ať už je tento rozsah oprávnění nadefinován staticky nebo dynamicky - vyhodnocen systémem v okamžiku, kdy se uživatel přihlásí k systému. Algoritmus je popsán od okamžiku, kdy má systém identifikovaného (autentifikovaného) uživatele, a zjistil množinu uživatelských míst, na kterých je uživatel zařazen. Vzhledem k tomu, že je povoleno, aby jeden uživatel byl přiřazen k více uživatelským místům, je třeba sestavit výslednou množinu povolených kombinací Osob/Výkazů (pro data), resp. výslednou množinu Výkazů (pro metadata) tak, že se zohlední nastavení rozsahu oprávnění přes všechna uživatelská místa, na která je uživatel přiřazen.

2.9.12.1 Typ oprávnění „metadata“

Cílem následujícího algoritmu je popsat proces vzniku seznamu Výkazů, ke kterým má mít uživatel přístup při projektování. Následující algoritmus je popsán za předpokladu, že se rozsah oprávnění vyhodnocuje pro jedno konkrétní uživatelské místo (v reálném provozu bude aplikován tolikrát, na kolika uživatelských místech je uživatel přiřazen). Výstupem algoritmu jsou dvě množiny Výkazů:

- výkazy, ke kterým má uživatel povolen přístup na základě statické nebo dynamické definice rozsahu oprávnění,
- výkazy, ke kterým nemá uživatel povolen přístup na základě negativních výjimek z dynamické definice rozsahu oprávnění.

Celkový rozsah oprávnění pro konkrétního uživatele je třeba vyhodnotit za všechna uživatelská místa, kam je uživatel platně zařazen. Tento algoritmus je tak vlastně sub-algoritmem celkového algoritmu vyhodnocování oprávnění popsaného v kapitole 2.9.13 Postup získání a vyhodnocení oprávnění.

Proces vyhodnocení rozsahu oprávnění probíhá následovně:

1. Systém načte pro konkrétní uživatelské místo seznam všech instancí objektu Rozsah oprávnění, které jsou k okamžiku vyhodnocování rozsahu oprávnění časově platné (atributy platnost_od/platnost_do objektu Rozsah oprávnění) a které jsou typu metadata (hodnota atributu „typ_oprávnění = metadata“). Vznikne tak množina rozsahu oprávnění pro uživatelské místo, označme ji „*RO*“.
2. Pro každý prvek množiny „*RO*“ (pro každou instanci objektu Rozsah oprávnění) systém vyhodnotí definici rozsahu oprávnění, který je definován staticky, a to takto:

a) z celkové množiny rozsahu oprávnění „RO“ získaných v bodě 1. vezme pouze ty prvky, pro které platí, že strana Výkazů je definována staticky, tedy hodnota atributu „**způsob_definice_výkazů = staticky**“. Vznikne množina rozsahu oprávnění „*RO_VYKAZ_STAT*“. Systém následně vytvoří prázdnou množinu výkazů, ke kterým je uděleno oprávnění pro projektování na základě statické definice rozsahu oprávnění. Množina bude označena „*VYK_STAT*“. Následně pro každý prvek množiny „*RO_VYKAZ_STAT*“ systém:
   i) načte všechny instance objektu Výkaz související s danou instancí objektu Rozsah oprávnění,
   ii) přidá do množiny „*VYK_STAT*“ ty výkazy, které získal v bodě i) a které v této množině zatím nejsou,
   iii) poté, co systém projde všechny staticky definované rozsahy oprávnění, vznikne množina výkazů, ke kterým má uživatel přístup pro projektování. Pokud se stane, že jeden a ten samý Výkaz bude v rámci různých instancí povolen vícekrát, ve výsledném seznamu se objeví právě jednou,

b) z celkové množiny rozsahu oprávnění „RO“ získaných v bodě 1. vezme pouze ty prvky, pro které platí, že strana Výkazů je definována dynamicky, tedy hodnota atributu „**způsob_definice_výkazů = dynamicky**“. Vznikne množina rozsahu oprávnění „*RO_VYKAZ_DYNAM*“. Systém následně vytvoří prázdnou množinu výkazů, ke kterým je uděleno oprávnění pro projektování na základě dynamické definice rozsahu oprávnění. Množina bude označena „*VYK_DYNAM_PLUS*“. Zároveň vytvoří prázdnou množinu „*VYK_DYNAM_MINUS*“, která bude určena pro uložení Výkazů, pro které byla definována výjimka z dynamického oprávnění. Následně pro každý prvek množiny „*RO_VYKAZ_DYNAM*“ systém:
   i) načte všechny instance objektu Vykazovací rámec související s instancí objektu Rozsah oprávnění,
   ii) načte všechny instance objektu Definice výjimky z rozsahu oprávnění související s instancí objektu Rozsah oprávnění,
   iii) pro každou zjištěnou instanci objektu Vykazovací rámec dle bodu i) systém načte seznam všech Výkazů, které jsou do daného vykazovacího rámce zařazeny, a toto zařazení je časově platné k okamžiku vyhodnocování rozsahu oprávnění (objekt „Výkaz ve Vykazovacím rámci“). Pravidlo časové platnosti lze potlačit nastavením atributu „ignorovat_datumovou_platnost“ v objektu „Rozsah oprávnění“ na hodnotu „true“),
   iv) pro každou zjištěnou instanci objektu Definice rozsahu oprávnění dle bodu ii) systém načte seznam všech Výkazů, pro které je definována výjimka,
   v) přidá do množiny „*VYK_DYNAM_PLUS*“ ty výkazy, které získal v bodě iii) a které v této množině zatím nejsou,
   vi) přidá do množiny „*VYK_DYNAM_MINUS*“ ty výkazy, které získal v bodě iv) a které v této množině zatím nejsou.

3. Systém pro uživatelské místo (tedy pro všechny prvky množiny „*RO*“):
   a) vygeneruje výsledný seznam výkazů, ke kterým má uživatel mít povolený přístup na základě statické i dynamické definice rozsahu oprávnění, a to tak, že posčítá prvky všech množin „*VYK_STAT*“ a „*VYK_DYNAM_PLUS“*, a zároveň z této množiny vyloučí případné duplicitní prvky. Vznikne množina „*UM_VYK_PLUS*“.
   b) výkazy, ke kterým nemá mít uživatel povolený přístup na základě výjimek z dynamické definice rozsahu oprávnění, a to tak, že **posčítá prvky všech množin**

„*VYK_DYNAM_MINUS***“,** které vznikly v bodě 2.b)vi) a zároveň z této množiny vyloučí případné duplicitní prvky. Vznikne množina „*UM_VYK_MINUS*“.

K výslednému určení toho, ke kterým Výkazům má uživatel přístup pro projektování je třeba výše uvedený postup aplikovat pro všechna uživatelská místa, ke kterým je uživatel přiřazen. Za každé uživatelské místo bude pomocí výše uvedeného algoritmu vrácen seznam Výkazů, ke kterým má uživatel povolen přístup a výkazů, ke kterým přístup povolen nemá.

V posledním kroku vyhodnocení oprávnění je nutno provést:

- distinctní sečtení všech prvků všech množin „*UM_VYK_PLUS*“. Vznikne množina „*USER_VYK_PLUS*“,
- distinctní sečtení všech prvků všech množin „*UM_VYK_MINUS*“. Vznikne množina „*USER_VYK_MINUS*“,
- odečtení všech prvků množiny „*USER_VYK_MINUS*“ od prvků množiny „*USER_VYK_PLUS*“.

Výsledek výše uvedené operace představuje seznam Výkazů, ke kterým má uživatel oprávnění pro projektování.

Výše uvedený postup má restriktivní charakter. Výjimky nejsou odečítány už na úrovni vyhodnocování za každý jeden rozsah oprávnění, ale až na úplném konci celého procesu vyhodnocování přístupových práv za všechna uživatelská místa, která jsou uživateli přiřazena. Tím je zajištěno, že jakákoli negativní výjimka vždy vyhraje nad dynamickou definici (nebo i statickou definicí). To znamená, že pokud je pomocí nějakého rozsahu oprávnění povolen přístup k výkazu „V123“ a v rámci jiného rozsahu oprávnění (klidně v rámci jiného uživatelského místa) je přístup k Výkazu „V123“ zakázán, pak je tento konflikt vyřešen „restriktivně“, **tedy přístup k Výkazu „V123“ bude uživateli odepřen.**

2.9.12.2 Typ oprávnění „data“

Cílem následujícího algoritmu je popsat proces vzniku seznamu kombinací osoba/výkaz, ke kterým má mít uživatel přístup při práci s Hodnotami údajů. Následující algoritmus je popsán za předpokladu, že rozsah oprávnění se vyhodnocuje pro jedno konkrétní uživatelské místo (v reálném provozu bude aplikován tolikrát, na kolika uživatelských místech je uživatel přiřazen). Výstupem algoritmu jsou dvě množiny Výkazů:

- množina kombinací Osoba/Výkaz, ke kterým má uživatel povolen přístup na základě statické nebo dynamické definice rozsahu oprávnění,
- množina kombinací Osoba/Výkaz, ke kterým nemá uživatel povolen přístup na základě výjimek z dynamické definice rozsahu oprávnění.

Proces vyhodnocení rozsahu oprávnění probíhá následovně:

1. Systém načte pro konkrétní uživatelské místo seznam všech instancí objektu Rozsah oprávnění, které jsou k okamžiku vyhodnocování oprávnění časově platné (atributy platnost_od/platnost_do objektu Rozsah oprávnění“) a které jsou typu data (hodnota atributu „typ_oprávnění = data“. Vznikne tak množina rozsahu oprávnění pro uživatelské místo, označme ji „*RO*“.

2. Pro každý prvek množiny „*RO*“ (pro každou instanci objektu Rozsah oprávnění) systém vyhodnotí definici rozsahu oprávnění, které jsou definovány staticky a to takto:
   a) z celkové množiny rozsahu oprávnění „RO“ získaných v bodě 1. vezme pouze ty prvky, pro které platí, že strana Výkazů je definována staticky, tedy hodnota atributu „**způsob_definice_výkazů = staticky** a strana Osob je definována taktéž staticky, tedy hodnota atributu „**způsob_definice_osob = staticky**“. Vznikne množina rozsahu oprávnění „*RO_STAT_STAT*“. Systém následně vytvoří prázdnou množinu výkazů. Množina bude označena „*VYKAZ_OSOBA_STAT_STAT*“. Následně pro každý prvek množiny „*RO_STAT_STAT*“ systém:
      i) načte všechny instance objektu Výkaz související s instancí objektu Rozsah oprávnění,
      ii) načte všechny instance objektu Osoba související s instancí objektu Rozsah oprávnění,
      iii) metodou kartézského součinu naplní množinu „*VYKAZ_OSOBA_STAT_STAT*“ všemi kombinacemi prvků získaných v bodě i) (Výkazy) a ii) (Osoby),
   b) z celkové množiny rozsahu oprávnění „RO“ získaných v bodě 1. vezme pouze ty prvky, pro které platí, že strana Výkazů je definována staticky, tedy hodnota atributu „**způsob_definice_výkazů = staticky**“ a strana Osob je definována dynamicky, tedy hodnota atributu „**způsob_definice_osob = dynamicky**“. Vznikne množina rozsahu oprávnění „*RO_STAT_DYNAM*“. Systém následně vytvoří prázdnou množinu kombinací výkaz/osoba. Množina bude označena „*VYKAZ_OSOBA_STAT_DYNAM*“. Zároveň systém vytvoří prázdnou množinu kombinací výkaz/osoba, do které bude ukládat výjimky z dynamicky definovaných rozsahů oprávnění pro stranu Osoba. Množina bude označená „*VYKAZ_OSOBA_STAT_DYNAM_VYJ*“. Následně pro každý prvek množiny „*RO_STAT_DYNAM*“ systém:
      i) načte všechny instance objektu Výkaz související s instancí objektu Rozsah oprávnění,
      ii) načte všechny instance objektu Typ osoby související s instancí objektu Rozsah oprávnění,
      iii) načte všechny instance objektu Definice výjimky z rozsahu oprávnění a zjistí všechny kombinace výkaz/osoba, pro které je udělena výjimka z dynamické definice rozsahu oprávnění,
      iv) pro každou zjištěnou instanci objektu Typ osoby dle bodu ii) systém načte seznam všech Osob, které jsou do daného typu osoby zařazeny, a toto zařazení je časově platné k okamžiku vyhodnocování rozsahu oprávnění (pravidlo časové platnosti lze potlačit nastavením atributu „ignorovat_datumovou_platnost“ v objektu „Rozsah oprávnění“ na hodnotu „true“),
      v) metodou kartézského součinu naplní množinu „*VYKAZ_OSOBA_STAT_DYNAM*“ všemi kombinacemi prvků získaných v bodě i) (Výkazy) a iv) (Osoby),
      vi) naplní množinu „*VYKAZ_OSOBA_STAT_DYNAM_VYJ*“ kombinacemi výjimek získanými v bodě iii),
   c) z celkové množiny rozsahu oprávnění „RO“ získaných v bodě 1. vezme pouze ty prvky, pro které platí, že strana Výkazů je definována dynamicky, tedy hodnota atributu „**způsob_definice_výkazů = dynamicky**“, a strana Osob je definována staticky, tedy hodnota atributu „**způsob_definice_osob = staticky**“. Vznikne množina rozsahu oprávnění „*RO_DYNAM_STAT*“. Systém následně vytvoří prázdnou množinu kombinací výkaz/osoba. Množina bude označena „*VYKAZ_OSOBA_DYNAM_STAT*“.

Zároveň systém vytvoří prázdnou množinu kombinací výkaz/osoba, do které bude ukládat výjimky z dynamicky definovaných oprávnění pro stranu Výkaz. Množina bude označena „*VYKAZ_OSOBA_DYNAM_STAT_VYJ*“. Následně pro každý prvek množiny „*RO_STAT_DYNAM*“ systém:

i) načte všechny instance objektu Vykazovací rámec související s instancí objektu Rozsah oprávnění,
ii) načte všechny instance objektu Osoba související s instancí objektu Rozsah oprávnění,
iii) načte všechny instance objektu Definice výjimky z rozsahu oprávnění a zjistí všechny kombinace výkaz/osoba, pro které je udělena výjimka z dynamické definice rozsahu oprávnění,
iv) pro každou zjištěnou instanci objektu Vykazovací rámec dle bodu i) systém načte seznam všech Výkazů, které jsou do daného vykazovacího rámce zařazeny a toto zařazení je časově platné k okamžiku vyhodnocování rozsahu oprávnění (objekt „Výkaz ve Vykazovacím rámci“). Pravidlo časové platnosti lze potlačit nastavením atributu „ignorovat_datumovou_platnost“ v objektu „Rozsah oprávnění“ na hodnotu „true“).
v) metodou kartézského součinu naplní množinu „*VYKAZ_OSOBA_DYNAM_STAT*“ všemi kombinacemi prvků získaných v bodě iv) (Výkazy) a ii) (Osoby),
vi) naplní množinu „*VYKAZ_OSOBA_DYNAM_STAT_VYJ*“ kombinacemi výjimek získanými v bodě iii),

d) z celkové množiny rozsahu oprávnění „RO“ získaných v bodě 1. vezme pouze ty prvky, pro které platí, že strana Výkazů je definována dynamicky, tedy hodnota atributu „**způsob_definice_výkazů = dynamicky**“ a strana Osob je definovaná taktéž dynamicky, tedy hodnota atributu „**způsob_definice_osob = dynamicky**“. Vznikne množina rozsahu oprávnění „*RO_DYNAM_DYNAM*“. Systém následně vytvoří prázdnou množinu kombinací výkaz/osoba. Množina bude označena „*VYKAZ_OSOBA_DYNAM_DYNAM*“. Zároveň systém vytvoří prázdnou množinu kombinací výkaz/osoba, do které bude ukládat výjimky z dynamicky definovaných oprávnění pro kombinaci výkaz/osoba. Množina bude označena „*VYKAZ_OSOBA_DYNAM_DYNAM_VYJ*“. Následně pro každý prvek množiny „*RO_DYNAM_DYNAM*“ systém:

i) načte všechny instance objektu Vykazovací rámec související s instancí objektu Rozsah oprávnění,
ii) načte všechny instance objektu Typ Osoby související s instancí objektu Rozsah oprávnění,
iii) načte všechny instance objektu Definice výjimky z rozsahu oprávnění a zjistí všechny kombinace výkaz/osoba, na které je udělena výjimka z dynamické definice rozsahu oprávnění,
iv) pro každou zjištěnou instanci objektu Vykazovací rámec dle bodu i) systém načte seznam všech Výkazů, které jsou do daného vykazovacího rámce zařazeny, a toto zařazení je časově platné k okamžiku vyhodnocování rozsahu oprávnění. Pravidlo časové platnosti lze potlačit nastavením atributu „ignorovat_datumovou_platnost“ v objektu „Rozsah oprávnění“ na hodnotu „true“).
v) pro každou zjištěnou instanci objektu Typ osoby dle bodu ii) systém načte seznam všech Osob, které jsou do daného typu osoby zařazeny, a toto zařazení je časově platné k okamžiku vyhodnocování rozsahu oprávnění,

vi) metodou kartézského součinu naplní množinu „*VYKAZ_OSOBA_DYNAM_DYNAM*“ všemi kombinacemi prvků získaných v bodě iv) (Výkazy) a v) (Osoby),

vii) metodou kartézského součinu naplní množinu „*VYKAZ_OSOBA_DYNAM_DYNAM_VYJ*“ kombinacemi výjimek získanými v bodě iii).

3. Systém pro uživatelské místo (tedy pro všechny prvky množiny „*RO*“):
   a) vygeneruje výsledný seznam kombinací Výkaz/Osoba, ke kterým má uživatel mít povolen přístup na základě statické i dynamické definice rozsahu oprávnění, a to tak, že posčítá prvky všech množin „*VYKAZ_OSOBA_STAT_STAT*“, „*VYKAZ_OSOBA_DYNAM_STAT“*, „*VYKAZ_OSOBA_STAT_DYNAM“* a *VYKAZ_OSOBA_DYNAM_DYNAM* a zároveň z této množiny vyloučí případné duplicitní prvky. Vznikne množina „*UM_VYKAZ_OSOBA_PLUS*“,
   b) vygeneruje výsledný seznam kombinací výkaz/osoba, ke kterým nemá mít uživatel povolen přístup na základě definice výjimek, a to tak, že posčítá prvky všech množin „*VYKAZ_OSOBA_STAT_DYNAM_VYJ*“, „*VYKAZ_OSOBA_DYNAM_STAT_VYJ“*, „*VYKAZ_OSOBA_DYNAME_DYNAM_VYJ“* a zároveň z této množiny vyloučí případné duplicitní prvky. Vznikne množina „*UM_VYKAZ_OSOBA_MINUS*“.

K výslednému určení toho, ke kterým kombinacím Osoba/Výkaz má uživatel přístup z hlediska vykázaných dat, je třeba výše uvedený postup aplikovat pro všechna uživatelská místa, ke kterým je uživatel přiřazen. Za každé uživatelské místo bude pomocí výše uvedeného algoritmu vrácen seznam kombinací osoba/výkaz, ke kterým má uživatel povolen přístup, a ke kterým kombinacím přístup povolen nemá.

V posledním kroku vyhodnocení oprávnění je nutno provést:

- distinctní sečtení všech prvků všech množin „*UM_VYKAZ_OSOBA_PLUS*“. Vznikne množina „*USER_VYKAZ_OSOBA_PLUS*“,
- distinctní sečtení všech prvků všech množin „*UM_VYKAZ_OSOBA_MINUS*“. Vznikne množina „*USER_VYKAZ_OSOBA_MINUS*“,
- odečtení všech prvků množiny „*USER_VYKAZ_OSOBA_MINUS*“ od prvků množiny „*USER_VYKAZ_OSOBA_PLUS*“.

Výsledek výše uvedené operace představuje seznam kombinací osoba/výkaz, ke kterým má uživatel právo pro čtení vykázaných dat.

Výše uvedený postup má restriktivní charakter. Negativní výjimky nejsou odečítány už na úrovni vyhodnocování za každý jeden rozsah oprávnění, ale až na úplném konci celého procesu vyhodnocování oprávnění. Tím je zajištěno, že negativní výjimka bude vždy upřednostněna. To znamená, že pokud je pomocí nějakého rozsahu oprávnění povolen přístup k výkazu „V123“ za osobu „O123“ a v rámci jiného rozsahu oprávnění (i v rámci jiného uživatelského místa) je přístup k Výkazu „V123“ za osobu „O123“ zakázán, pak je tento konflikt vyřešen „restriktivně“, **tedy uživatel přístup k Výkazu „V123“ za osobu „O123“ bude uživateli odepřen.**

2.9.13 Postup získání a vyhodnocení oprávnění

Cílem tohoto algoritmu je popsat algoritmus vzniku oprávnění uživatele. Oprávnění je složeno z definice přístupových práv (role a aktivity; CO uživatel smí v systému dělat) a z definice rozsahu oprávnění (S ČÍM – s jakými daty – to smí dělat).

Systém každému uživateli, který prokáže svoji identitu (viz kapitoly 3.4 Popis procesu autentizace — externí registrovaný uživatel a 3.5 Popis procesu autentizace — interní uživatel), nastaví oprávnění dle následujícího algoritmu:

1. Systém zjistí množinu všech aktivních a k okamžiku přihlášení časově platných uživatelských míst, na kterých je uživatel přiřazen (viz kapitoly 2.2 Objekt Uživatel, 2.5 Objekt Uživatelské místo. V případě, že uživatel použil volbu, která mu umožňuje pracovat pouze pod jedním uživatelským místem, pak je tato množina tvořena právě tímto jedním uživatelským místem (viz UMU_19.0).
2. Systém pro každé uživatelské místo z množiny vytvořené podle bodu 1.:
   a) načte seznam všech souvisejících, aktivních a k okamžiku přihlášení časově **platných rolí**, které jsou k danému uživatelskému místu připojeny (viz kapitoly 2.6 Objekt Role). Vznikne množina rolí a ke každé roli z této množině systém načte seznam všech přiřazených aktivních a časově **platných aktivit** (viz kapitoly 2.7 Objekt Aktivita). Vznikne tak množina aktivit a ta může v tuto chvíli obsahovat duplicitní aktivity. Systém tyto duplicity vyloučí a ponechá vždy jen jednu aktivitu (distinctní výběr instancí objektu Aktivita),
   b) načte seznam všech souvisejících, aktivních a k okamžiku přihlášení časově platných instancí objektu **Rozsah oprávnění**. Přesný popis vyhodnocení definice rozsahu oprávnění je popsán v kapitole 2.9.12.1 Typ oprávnění „metadata“ pro oblast projektování výkazů a v kapitole 2.9.12.2 Typ oprávnění „data“ pro oblast přístupu k vykázaným Hodnotám údajů.
3. Pro každé jedno uživatelské místo z množiny pod bodem 1. systém na základě popisu uvedeném v bodu 2. vytvoří právě jedno oprávnění. Oprávnění je tak vlastně kombinace množiny aktivit (co smí uživatel provádět) a množiny výkazů (metadata) nebo množiny kombinací osoba/výkaz (data). Následující pravidla budou aplikována při vytváření oprávnění:
   a) pokud v rámci daného uživatelského místa, pro které se sestavuje oprávnění, není k dispozici žádná aktivita, pak systém oprávnění nevytvoří (nelze definovat CO má uživatel povoleno a povolit VŠECHNY dostupné aktivity je nežádoucí),
   b) pokud v rámci daného uživatelského místa, pro které se sestavuje oprávnění, není k dispozici žádný rozsah oprávnění, pak systém:
      i) u interních uživatelských míst nevytvoří oprávnění,
      ii) u externích uživatelských míst nastaví rozsah oprávnění právě na jednu Osobu (externí uživatelské místo musí být spjato s právě jednou Osobou) a na všechny Výkazy, ke kterým je pro danou Osobu definována vykazovací povinnost (bez ohledu na datum, ke kterému je tato vykazovací povinnost definována).
4. Oprávnění není reprezentováno žádným reálným fyzickým objektem sytému, je to za běhu systému sestavená kombinace instancí několika objektů (Role/Aktivita vs. Rozsah oprávnění), která je sestavena vždy v okamžiku přihlášení uživatele do systému.

Systém následně dynamicky povoluje/zakazuje funkcionality (funkcionalitou se rozumí systémová aktivita) na základě získaných oprávnění v bodě 4. Funkcionalita systému

(systémová aktivita) je uživateli povolena, pokud je s ní související aplikační aktivita v seznamu oprávnění v kombinaci s právě vybranou Osobou a Výkazem. Tam, kde se daná funkcionalita nevztahuje k Výkazu/Osobě, není nutno vyhodnocovat část oprávnění týkající se výkazu/osoby. Pro lepší pochopení rozdílu mezi aplikační a systémovou aktivitou viz kapitoly 2.7 Objekt Aktivita (Aplikační aktivita) a 2.8 Objekt Systémová aktivita.

2.9.14 Speciální definice rozsahu oprávnění

Kromě definice rozsahu oprávnění přes objekt Rozsah oprávnění, která opisuje přístupy na vybrané instance objektu Výkaz nebo kombinaci instancí osoba/výkaz, existuje další způsob definice práv pro přístup k instancím nějakého objektu.

2.9.14.1 Objekt Číselník

V rámci objektu Číselník rozlišujeme, zda se jedná o číselník globální nebo číselník lokální:

- **globální číselník** – jedná se o číselník, který je používán v širokém spektru výkazů a je žádoucí, aby jakékoli změny v obsahu takového číselníku byly prováděny centrálně a byly koordinovány přes více uživatelů. Typickým příkladem takového číselníku je číselník zemí nebo národních měn. Obsah globálních číselníků není dovoleno měnit autorům výkazů, i když daný číselník ve svých výkazech používají,
- **lokální číselník** – jedná se o číselník, který je specifický buď pro jeden výkaz anebo pro velmi úzkou množinu výkazů. U lokálních číselníků je požadováno, aby jejich editaci mohl provádět autor výkazu, protože se má za to, že změny v lokálním číselníku neovlivňují ostatní uživatele (projektanty výkazů) na rozdíl od změn prováděných v globálních číselnících.

Objekt Číselník obsahuje atribut „atribut_pro_správu_objektu“, který nabývá hodnot „lokální“ a „globální“, viz dokument B – Metapopis. Zároveň existují následující aktivity:

- vytvořit/změnit/smazat globální číselník a vytvořit/změnit/smazat položky globálního číselníku,
- vytvořit/změnit/smazat lokální číselník a vytvořit/změnit/smazat položky lokálního číselníku.

Zároveň existuje i aktivita „číst číselník“; zde se však již nerozlišuje, zda se jedná o číselník lokální nebo globální. Uživatel, který má přiřazenu aktivitu „číst číselník“, může číst instance objektu Číselník a instance objektu Položky číselníku (nelze oddělit právo na čtení pro objekt Číselník a pro objekt Položky číselníku), tj. má právo číst jakýkoli číselník (a jeho obsah) bez ohledu na to, zda je globální nebo lokální.

Aktivity týkající se globálních číselníků budou přiřazeny uživatelům s právem spravovat tento druh číselníků. Úkolem takových uživatelů je zajistit to, aby se číselníky v systému neduplikovaly a bylo tak možno udržet datovou konzistenci vykazovaných dat. Zároveň platí, že ten, kdo má právo vytvořit globální číselník, má právo potvrdit lokální číselník.

Aktivity týkající se lokálních číselníků jsou pak přiřazeny těm uživatelům, kteří mají mít právo vytvářet a modifikovat lokální číselníky. Pokud uživatel má takové právo, pak může vytvořit lokální číselník. Lokální číselník však začne platit až po schválení uživatelem, který

k tomu má patřičné oprávnění. Podrobně je celý proces tvorby a schvalování lokálního číselníku popsán v dokumentu B – Metapopis.

2.10 Vyhodnocení přístupových práv pro uživatelské pohledy

Jedním z požadavků na systém SDAT jsou tzv. uživatelské pohledy (viz dokument F – Výběry dat, kapitola 5 Uživatelské pohledy). Uživatelské pohledy je sada databázových view. Uživatelé s patřičnými právy následně mohou použít jakýkoli externí nástroj pro připojení k databázi (MS Excel, MS Access, TOAD, SQL Developer atd.) a přistupovat k datům mimo systém SDAT.

Uživatelské pohledy jsou dostupné výhradně interním uživatelům.

Pokud však bude systém SDAT realizován pomocí tzv. třívrstvé architektury, pak v pro přímý přístup k uživatelským pohledům platí, že tento přístup nebude vykonán přes aplikační server a tudíž nelze provést standardní proces vyhodnocení přístupových práv, tak jak je popsán v případě, že uživatel k datům v rámci práce s aplikací.

Vytvoření databázového účtu uživateli, který má mít přístup k uživatelským pohledům je v kompetenci ČNB a proběhne tak, že uživatel bude zařazen do příslušné aplikační skupiny v Řídicí databázi. Tato akce proběhne mimo systém SDAT.

Jakmile má uživatel vytvořený databázový účet, pak je nutné, aby se tento databázový účet zanesl k reálnému uživateli systému SDAT (atribut „DB účet interního uživatele“, viz objekt Uživatel). V okamžiku, kdy se tak stane, pak systém musí provést „grantování“ přístupu k jednotlivým view pro daný databázový účet a to podle následujícího algoritmu:

- Systém zjistí aktuálně platnou množinu oprávnění daného uživatele.
- Systém udělí grant „select“ na ta view, které reprezentují ty výkazy, pro které platí, že daný uživatel má právo na všechny Osoby pro daný výkaz (view je vždy koncipováno tak, aby pokrývalo právě jeden výkaz).

Dále je nutné, aby systém SDAT prováděl při každé změně definice rozsahu oprávnění tyto činnosti:

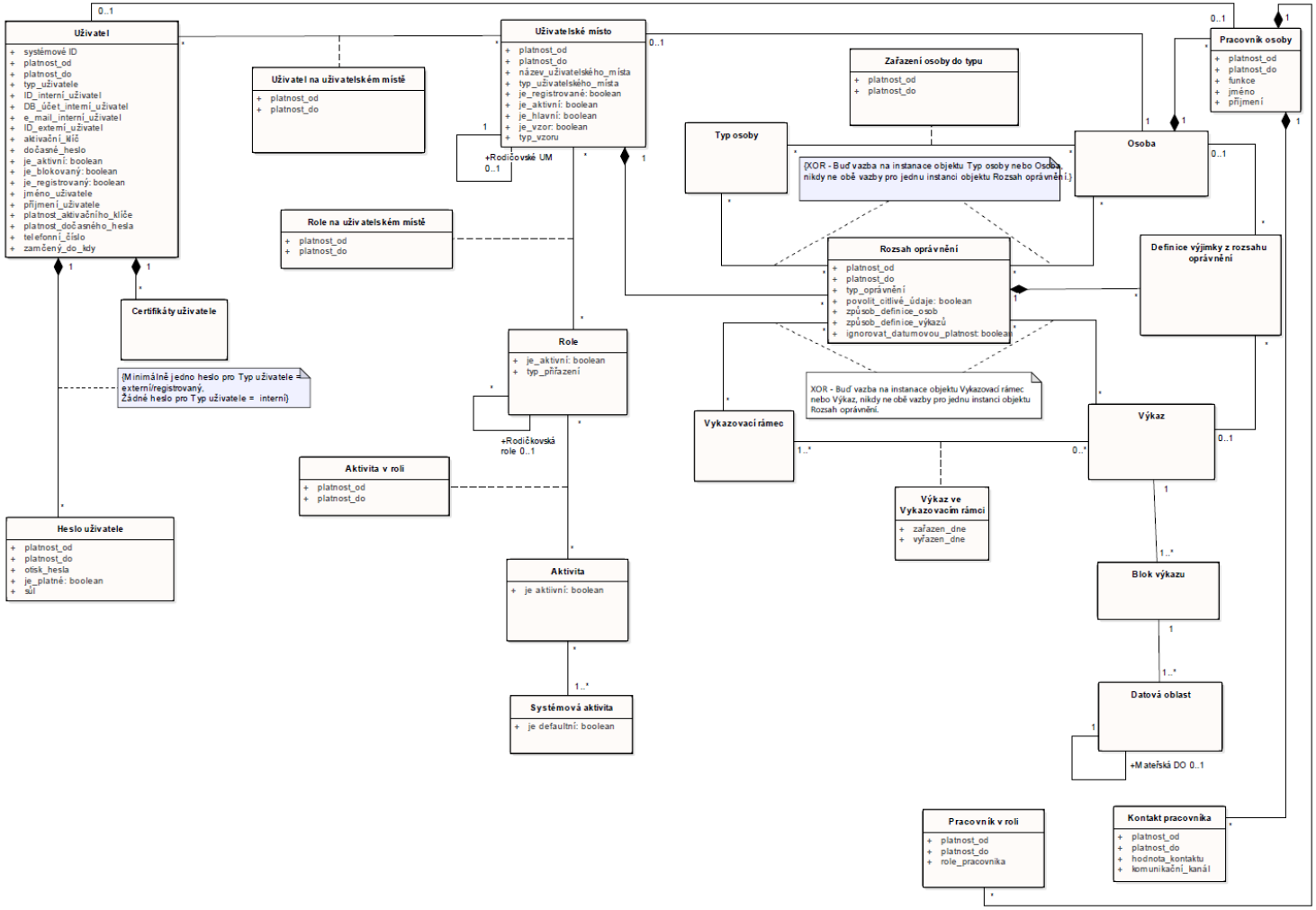
- Přidal právo „select“ (grant) těm uživatelům, u kterých po změně oprávnění došlo k tomu, že nově začala platit podmínka, že mají, pro daný výkaz, právo na všechny Osoby
- Odebral právo „select“ (revoke) těm uživatelům, u kterých po změně oprávnění došlo k tomu, že přestala platit podmínka, že mají, pro daný výkaz, právo na všechny Osoby.
- Přidal právo „select“ (grant) těm uživatelům, u kterých po vytvoření nového view, je splněna podmínka, že mají právo, pro daný výkaz, na všechny Osoby

2.11 Objekt Definice výjimky z rozsahu oprávnění

Objekt Definice výjimky z rozsahu oprávnění slouží k zachycení výjimek z dynamicky definovaného rozsahu oprávnění. Podrobně je tento objekt a způsob jeho využití popsán v kapitolách, které se zabývají vyhodnocením oprávnění – viz kapitoly 2.9.7 Rozsah

oprávnění – možnosti definice pro oblast „metadata“, 2.9.8 Rozsah oprávnění – možnosti definice pro oblast „data“ a 2.9.12 Vyhodnocení definice rozsahu oprávnění.

2.12 Objektový model pro oblast Uživatelé a Oprávnění



Obrázek 1 - Objektový model pro oblast Uživatelé a Oprávnění

3 Procesy

3.1 Popis procesu vytvoření uživatelského účtu uživatelem v ČNB

3.1.1 Účel procesu

Základním účelem procesu je vytvořit v ČNB pro uživatele externí Osoby nebo pro interního uživatele alespoň jeden aplikační účet uživatele, pomocí kterého bude uživatel provádět aktivity v systému SDAT, a pro tento aplikační účet uživatele bezpečně vytvořit uživateli heslo, které bude známo pouze jemu. Proces řeší vznik uživatelského aplikačního účtu pro interní uživatele ČNB a externí registrované uživatele. Neřeší proces vzniku aplikačního účtu uživatele formou autoregistrace, tento proces je popsán v kapitole 3.7 Popis procesu vytvoření externí Osoby a aplikačního účtu uživatele externím subjektem (Autoregistrace).

3.1.2 Výchozí situace

Do ČNB byla bezpečnou cestou (doporučeným dopisem splňující všechny právní povinnosti, zprávou do datové schránky, e-mailem podepsaným uznávaným elektronickým podpisem) doručena žádost o vytvoření nové Osoby a informace o tom, jaké aplikační účty mají být pro danou osobu založeny. Tato žádost musí obsahovat zejména:

- jméno a příjmení uživatele, kterému má být zřízen aplikační účet uživatele,
- e-mail daného uživatele,
- nepovinně může být připojena informace o elektronickém podpisu (kvalifikovaném certifikátu) uživatele[4]:
    - ID certifikační autority, která certifikát vydala,
    - SerialNumber.
    - Datum platnosti certifikátu

Identitu externího uživatele ověřuje systém SDAT zadáním jedinečné kombinace uživatelské jméno/heslo. Kvalifikovaný certifikát není považován za doklad pro ověření identity uživatele, ale pouze pro tzv. „elektronický podpis“ (viz http://www.mvcr.cz/clanek/informace-k-pouzivani-kvalifikovanych-certifikatu-pro-elektronicky-podpis-a-zaroven-pro-autentizaci-a-sifrovani.aspx).

Správci aplikace je dohodnutým způsobem doručen požadavek na vytvoření nového interního aplikačního účtu uživatele.

[4] Kvalifikovaný certifikát bude uživatel potřebovat jen pro zaslání pouze některých výkazů. Jeho poskytnutí tak není povinné. Pokud se bude jednat o subjekt, který zasílá jen Výkazy, které není nutné podepisovat kvalifikovaným certifikátem, nemusejí být údaje o certifikátu předány.

3.1.3 Spouštěč procesu

Schválený a ověřený požadavek na vytvoření nového aplikačního účtu uživatele pro externí Osobu nebo pro interního uživatele ČNB.

3.1.4 Průběh procesu

3.1.4.1 Hlavní úspěšný scénář

Hlavní úspěšný scénář popisuje vytvoření nového aplikačního účtu uživatele a jeho zařazení na uživatelské místo (přidělení oprávnění). V případě, že se jedná o vytvoření uživatelského aplikačního účtu uživatele pro externího uživatele, pak aktivity popsané v tomto scénáři probíhají vždy v kontextu právě jedné Osoby (předpokladem scénáře je, že Osoba již existuje).

Proces probíhá následovně:

1. Uživatel zadá následující údaje nutné pro vytvoření aplikačního účtu uživatele:
   a) jméno uživatele,
   b) příjmení uživatele,
   c) e-mailovou adresu uživatele,
   d) typ uživatele (interní/externí),
   e) alias uživatele (interní uživatelské jméno; pouze v případě interních aplikačních účtů uživatele).
2. Systém provede kontrolu validity zadaných dat a ověří, zda předaná e-mailová adresa je validní e-mailovou adresou[5] a zda je evidována u nějakého jiného aplikačního účtu uživatele a zda alias uživatele (interní uživatelské jméno) je evidováno u nějakého jiného aplikačního účtu uživatele (pouze pro typ uživatele = „interní“). Pokud:
   a) systém zjistí, že předaná e-mailová adresa není validní, označí dané pole v zadávacím formuláři, vypíše chybovou zprávu a neumožní aplikační účet uživatele vytvořit, dokud uživatel chybu neopraví,
   b) systém zjistí, že e-mailová adresa je již v systému použita u jiného aplikačního účtu uživatele, neumožní aplikační účet uživatele vytvořit (pro kontrolu duplicity existence e-mailové adresy jsou vyloučeny ty instance objektu Uživatel, které jsou určeny pro neregistrovaný přístup, tedy kde je atribut instance objektu Uživatel „je_registrovaný = ne“). V případě, že je e-mailová adresa uživatele shledána jako duplicitní, systém o této skutečnosti informuje uživatele chybovým hlášením a scénář končí,
   c) systém pro „typ uživatele = interní“ zjistí, že alias uživatele (interní uživatelské jméno) je již v systému použit u jiného aplikačního účtu uživatele, neumožní aplikační účet uživatele vytvořit. V případě, že je alias uživatele shledán jako duplicitní, systém o této skutečnosti informuje uživatele chybovým hlášením a scénář končí.
3. Systém vytvoří nový aplikační účet uživatele (instanci objektu Uživatel) dle následujících pravidel:
   a) jako ID uživatele je použita e-mailová adresa uživatele zadaná v kroku 1.c),

[5] Validita e-mailové adresy bude v této fázi ověřena pomocí regulárního výrazu.

b) další atributy aplikačního účtu uživatele budou nastaveny takto:
   i) v případě, že je atribut „typ_uživatele“ = interní“, pak je nutno zadat atribut „alias_uživatele“. Alias uživatele je interní jméno uživatele v rámci domény ČNB. Hodnota atributu „alias_uživatele“ musí být jedinečná přes všechny instance objektu Uživatel,
   ii) atribut „je_aktivní“ má hodnotu „ne“,
   iii) atribut „aktivační_klíč“ obsahuje systémem náhodně vygenerovaný a přes celou databázi jedinečný řetězec o délce minimálně 24 znaků,
   iv) atribut „platnost_aktivačního_klíče“ je nastavena na datum a čas, které odpovídá okamžiku vytvoření aplikačního účtu uživatele + 168 hodin,
   v) atribut „je_registrovaný“ má hodnotu „ano“,
   vi) atribut „je_blokovaný“ má hodnotu „ne“,
   vii) atribut „zamčený_do_kdy“ nemá žádnou hodnotu,
   viii) atributy „jméno_uživatele“ a „příjmení_uživatele“ zadá uživatel a systém neumožní ponechat tyto údaje bez vyplnění,
   ix) atribut „mobilní telefonní_číslo“ je pro interní a externí registrované uživatele nepovinný.

4. Systém volitelně umožní uživateli k vytvářenému aplikačnímu účtu uživatele přiřadit certifikát (vytvoření instance objektu Certifikáty uživatele). Proces přiřazení certifikátu je popsán v kapitole 3.1.4.2 Subproces Vložení informací o elektronickém podpisu.
5. Systém pouze pro „typ uživatele = externí“ na zadanou e-mailovou adresu odešle e-mailovou zprávu, která bude obsahovat informace nutné pro aktivaci účtu, viz kapitola 3.1.4.3 Subproces Odeslání aktivačního e-mailu externímu registrovanému uživateli.
6. Externí uživatel provede aktivaci účtu. Systém umožňuje uživateli provést aktivaci neaktivního aplikačního účtu uživatele dvěma způsoby:
   a) zadáním ID uživatele a aktivačního klíče uživatelem do aktivačního dialogu,
   b) předáním URL, která obsahuje informaci o ID uživatele a o aktivačním klíči.
7. Před samotným provedením aktivace aplikačního účtu uživatele systém provede kontrolu:
   a) zda k aktivaci nedochází po uplynutí platnosti aktivačního klíče (aktuální datum a čas je větší než atribut „platnost_aktivačního_klíče“). Pokud ano, systém aktivaci neprovede a informuje uživatele o tom, že platnost aktivačního klíče vypršela a je nutné celý proces založení aplikačního účtu uživatele opakovat,
   b) zda existuje aplikační účet uživatele, pro který probíhá aktivace (pokud uplynula doba určená na aktivaci, systém aplikační účet uživatele mezitím smazal). Pokud aplikační účet uživatele neexistuje, systém aktivaci neprovede a informuje uživatele o tom, že platnost aktivačního klíče vypršela a je nutné celý proces založení aplikačního účtu uživatele opakovat,
   c) zda aplikační účet uživatele, pro který probíhá aktivace je označen jako neaktivní (aktivační URL může být použita opakovaně). V případě, že je proveden pokus aktivovat aplikační účet uživatele, který již aktivován byl, systém aktivaci neprovede a přesměruje uživatele na přihlašovací dialog.
8. V případě, že jsou všechny podmínky pro aktivaci účtu splněny, pak systém požádá uživatele o vytvoření prvotního hesla. Proces vytvoření prvotního hesla je popsán v kapitole 3.1.4.4 Subproces Vytvoření prvotního hesla k aplikačnímu uživatelskému účtu.
9. V případě, že prvotní heslo uživatele bylo úspěšně vytvořeno (vznikla instance objektu Heslo uživatele) pak systém provede aktivaci aplikačního účtu uživatele následovně:

a) atribut „je_aktivní“ nastaví na hodnotu „ano“,
b) atribut „aktivační klíč“ nastaví na hodnotu NULL,
c) atribut „platnost_aktivačního_klíče“ nastaví na hodnotu NULL.

10. Systém dle typu uživatele:
    a) externího uživatele po dokončení aktivace účtu přesměruje na přihlašovací dialog,
    b) internímu uživateli odešle na e-mailovou adresu spojenou s daným aplikačním účtem informační e-mail. Součástí tohoto e-mailu budou tyto informace:
        i) URL, která umožní uživateli spustit aplikaci,
        ii) kontaktní údaje na uživatele ČNB, kteří mohou pomoci s případnými problémy při aktivaci aplikačního účtu uživatele.
11. **Systém vytvoří tzv. hlavní uživatelské místo**, kterému budou přiřazena taková oprávnění, aby bylo možno provádět vykazování a zakládání dalších aplikačních účtů v rámci Osoby. Toto uživatelské místo bude navázáno k Osobě, pro kterou je aplikační účet uživatele vytvářen.
12. Systém přiřadí k hlavnímu uživatelskému místu uživatele, kterého založil v bodě 3..

3.1.4.2 Subproces Vložení informací o elektronickém podpisu

V rámci procesu vytvoření instance objektu Uživatel (aplikačního účtu uživatele) systém uživateli volitelně umožní k vytvářenému aplikačnímu účtu uživatele přiřadit informace o elektronickém podpisu uživatele (informace o kvalifikovaném certifikátu). Uživatel bude používat elektronický podpis v případě, že bude odesílat Vstupní zprávu, která bude obsahovat Výkazy, u kterých bude předepsáno, že mají být podepsány elektronickým podpisem. Systém umožňuje vložení informací o elektronickém podpisu těmito způsoby:

- **zadáním názvu vystavitele a sériového čísla**. V takovém případě na základě znalosti této jedinečné kombinace údajů zajistí systém (s využitím interní služby CRT) stažení veřejné části certifikátu ve formátu PEM od patřičné autority pomocí veřejného aplikačního rozhraní dané autority,
- **zadáním veřejného klíče certifikátu ve formátu PEM.** V takovém případě systém (s využitím interní certifikační služby CRT) zjistí vystavitele certifikátu a sériové číslo certifikátu

V rámci vytváření instance objektu Certifikáty uživatele je vykonána kontrola získaných informací o certifikátu (elektronickém podpisu). Předmětem kontroly je:

- **platnost data certifikátu** (certifikát musí být platný v časovém okamžiku, kdy je zakládán aplikační účet uživatele),
- **platnost certifikátu** (certifikát nesmí být revokovaný).

V případě, že elektronický podpis nevyhoví alespoň jedné výše popsané kontrole, systém na tuto skutečnost upozorní uživatele a instanci objektu Certifikát uživatele nevytvoří. Na samotné vytvoření instance objektu Uživatele (aplikačního účtu uživatele) tato skutečnost nemá vliv (instance objektu Uživatel bude vytvořena, i když nebude vytvořena instance objektu Certifikáty uživatele).

Výše uvedená pravidla platí i pro vytváření instancí objektu Certifikáty uživatele i v případě, že jsou informace o elektronickém podpisu vkládány k již existujícímu uživateli.

3.1.4.3 Subproces Odeslání aktivačního e-mailu externímu registrovanému uživateli

Systém po dokončení procesu vytvoření aplikačního účtu uživatele externímu registrovanému uživateli odešle na e-mailovou adresu spojenou s daným aplikačním účtem aktivační e-mail. Součástí tohoto e-mailu budou tyto informace:

- o založení nového aplikačního účtu uživatele a nutnosti jej do jednoho týdne aktivovat (jinak bude smazán),
- URL, kde běží aplikace a kde a jak je možno provést aktivaci,
- ID uživatele, pomocí kterého se bude uživatel přihlašovat k systému,
- aktivační klíč, pomocí kterého bude provedena aktivace aplikačního účtu uživatele a informaci o datu a času, do kdy je aktivační klíč aktivní. Aktivační klíč je předán jednak samostatně a jednak formou URL (linku), která umožní přímou aktivaci aplikačního účtu uživatele bez nutnosti zadávat aktivační klíč,
- kontaktní údaje na uživatele ČNB, kteří mohou pomoci s případnými problémy při aktivaci aplikačního účtu uživatele.

3.1.4.4 Subproces Vytvoření prvotního hesla k aplikačnímu uživatelskému účtu

Systém umožňuje uživateli vytvořit prvotní heslo k aplikačnímu účtu uživatele následujícím způsobem:

1. Systém si od uživatele vyžádá zadání hesla do dvou různých polí (pole „heslo“ a pole „potvrzení hesla“)
2. Uživatel zadá do obou polí sadu znaků.
3. V případě, že jako heslo bude uživatelem zadán stejný řetězec znaků, jako je jeho uživatelské jméno, systém na tuto skutečnost upozorní uživatele a nebude pokračovat v procesu změny hesla do doby, kdy bude heslo odlišné od uživatelského jména.
4. Systém se pokusí heslo vytvořit. Aby to bylo možné, musí být splněny tyto podmínky:
   a) systém rozlišuje malá a velká písmena („a“ je jiný znak než „A“),
   b) zadané řetězce v obou polích musejí být shodné. Pokud se liší, systém informuje uživatele hlášením, že se zadaná hesla neshodují. Proces vytvoření hesla je ukončen a heslo není vytvořeno,
   c) zadaný řetězec v poli „heslo“ musí splňovat všechny podmínky dle aktuálně definované bezpečnostní politiky (viz kapitola 4.4 Definice bezpečnostní politiky). V případě, že nějaká podmínka/nějaké podmínky na heslo kladená není/nejsou splněna/y, systém uživatele informuje hlášením, jaká/jaké podmínka/y není/nejsou splněna/y a vyžádá si opravu zadání. Proces vytvoření hesla je ukončen a heslo není vytvořeno.

V případě, že nejsou splněny všechny nutné podmínky pro vytvoření hesla, systém heslo nevytvoří. Proces aktivace účtu není dokončen a účet zůstává neaktivní.

Systém po vytvoření nového hesla (vytvoření nové instance objektu Heslo uživatele) nastaví:

- hodnotu atributu „platnost_od“ na aktuální datum a čas,
- hodnota atributu „platnost_do“ na aktuální datum a čas a počet minut uvedených v aktuálně platné konfigurační položce **PWD_MAX_AGE**.

Heslo není do systému ukládáno v otevřené podobě, ale je šifrováno v rozsahu požadavků definovaných v kapitole 4.1 Systém uložení hesel.

3.1.5 Výstup procesu

Úspěšně vytvořený aplikační účet uživatele pro uživatele v rámci externí registrované Osoby.

3.2 Popis procesu změny hesla uživatele

Každý externí registrovaný uživatel může kdykoli sám ke svému aplikačnímu účtu uživatele změnit heslo, případně změna hesla může být vynucena systémem (v případě vypršení platnosti hesla).

3.2.1 Účel procesu

Zajištění plnění bezpečnostní politiky.

3.2.2 Výchozí situace

Existuje externí registrovaný uživatel, který disponuje aktivním aplikačním účtem uživatele.

3.2.3 Spouštěč procesu

Proces je spouštěn:

- akcí uživatele – uživatel se rozhodl, že si změní heslo ke svému aplikačnímu účtu uživatele,
- systémem – do systému SDAT se pokouší přihlásit externí registrovaný uživatel, kterému vypršela platnost hesla. V takovém případě systém SDAT vynutí změnu hesla.

3.2.4 Průběh procesu

Hlavní úspěšný scénář

Proces změny hesla probíhá následovně:

1. Uživatel je vyzván k zadání následujících údajů:
   a) aktivní heslo (poslední platné heslo bez ohledu na to, zda už vypršela jeho časová platnost nebo ne; objekt Heslo uživatele, atribut „je_platné = ano“),
   b) nové heslo,
   c) potvrzení nového hesla.
2. Uživatel zadá do obou polí sadu znaků.

3. V případě, že jako heslo bude uživatelem zadán stejný řetězec znaků jako je jeho uživatelské jméno, systém na tuto skutečnost upozorní uživatele a nebude pokračovat v procesu změny hesla do doby, kdy bude heslo odlišné od uživatelského jména.
4. Aby bylo možné změnit heslo, musí být splněny tyto podmínky:
   a) systém rozlišuje malá a velká písmena („a“ je jiný znak než „A“),
   b) zadané aktivní heslo musí odpovídat existujícímu platnému heslu (atribut „otisk_hesla“ u instance objektu Heslo uživatele, kde atribut „je_platné = ano“). Při porovnání hesel je třeba vzít v úvahu, že v databázi je uložena šifrovaná hodnota hesla a je tak nutné před porovnáním obou řetězců (heslo zadané uživatelem a heslo uložené v databázi) provést zašifrování zadaného hesla (viz kapitola 4.1 Systém uložení hesel). V případě, že uživatelem zadané aktivní heslo neodpovídá heslu uloženému v databázi, systém informuje uživatele hlášením, že zadané aktivní heslo neodpovídá aktivnímu heslu uloženému v databázi. Proces změny hesla je ukončen, heslo není změněno,
   c) zadané řetězce v obou polích „nové heslo“ a „potvrzení hesla“ musejí být shodné. Pokud se liší, systém informuje uživatele hlášením, že se zadaná hesla neshodují. Proces vytvoření hesla je ukončen, heslo není vytvořeno,
   d) zadaný řetězec v poli „nové heslo“ musí splňovat všechny podmínky dle aktuálně definované bezpečnostní politiky, viz kapitola 4.4 Definice bezpečnostní politiky, v případě, že nějaká podmínka/nějaké podmínky na heslo kladená není/nejsou splněna/y, systém uživatele informuje hlášením, jaká/jaké podmínka/y není/nejsou splněna/y a vyžádá si opravu zadání. Proces změny hesla je ukončen a heslo není změněno.
5. Systém změní heslo následujícím způsobem:
   a) stávající aktivní heslo (atribut „je_platné = ano“) bude ukončeno tak, že atribut „platnost_do“ bude nastaven na aktuální časový okamžik minus 1 sekunda. Toto platí pouze v případě, že je dané heslo časově platné. Pokud dochází ke změně hesla po jeho vypršení, nebude atribut „platnost_do“ nijak dotčen,
   b) stávající aktivní heslo bude označeno jako neplatné (atribut „je_platné“ = „ne“),
   c) bude vytvořena nová instance objektu Heslo uživatele:
      i) atribut „platnost_od“ je nastaven na hodnotu aktuálního data a času,
      ii) atribut „platnost_do“ je nastaven na hodnotu aktuálního data a času plus počet minut uvedených v aktuálně platné konfigurační položce **PWD_MAX_AGE**.

Heslo není do systému ukládáno v otevřené podobě, ale je šifrováno v rozsahu požadavků definovaných v kapitole 4.1 Systém uložení hesel.

3.2.5 Výstup procesu

Změněné heslo k aplikačnímu účtu uživatele.

3.3 Popis procesu resetování hesla uživatele

Každý externí registrovaný uživatel může kdykoli sám ke svému aplikačnímu účtu uživatele resetovat heslo. Resetováním hesla se rozumí postup, který umožňuje uživateli obnovit aktuálně platné heslo (má se na mysli i takové heslo, které již exspirovalo), které zapomněl.

3.3.1 Účel procesu

Snížení nároků na administraci systému, odstranění potřeby předávat uživateli vygenerované heslo při současném zajištění plnění bezpečnostní politiky.

3.3.2 Výchozí situace

Existuje externí registrovaný uživatel, který disponuje aktivním aplikačním účtem uživatele, ke kterému však zapomněl heslo. Možnost provést reset (obnovu) hesla je dostupná z přihlašovacího dialogu, tedy v části aplikace, která je veřejně přístupná.

3.3.3 Spouštěč procesu

Akce uživatele – uživatel během autentizačního procesu zjistil, že si již nepamatuje svoje heslo.

3.3.4 Průběh procesu

3.3.4.1 Hlavní úspěšný scénář

Proces obnovy hesla bude probíhat následovně:

1. Uživatel požádá o resetování zapomenutého hesla (volba „Resetovat zapomenuté heslo“ bude dostupná na přihlašovacím formuláři).
2. Systém si vyžádá zadání ID uživatele (e-mailovou adresu), pro který má být heslo resetováno.
3. Systém ověří, zda existuje aplikační účet uživatele s předaným ID externího uživatele:
   a) **pokud ano**, pak prověří, zda daný aplikační účet uživatele:
      i) je aktivní („je_aktivní = ano“),
      ii) jedná se o externí registrovaný aplikační účet uživatele („je_registrovaný = ano“ a zároveň „typ_uživatele = externí“),
      iii) není blokovaný („je_blokovaný = ne“),
      iv) není zamčený („zamčený_do_kdy = NULL“)
      v) nebylo pro něj dříve zažádáno o reset hesla a tento proces resetu nebyl dokončen (atribut „platnost_dočasného_hesla“ obsahuje hodnotu, která je vyšší než aktuální systémový datum a čas v okamžiku vznesení požadavku na reset hesla),

      pokud alespoň jedna z výše uvedených kontrol není splněna, systém resetování hesla neumožní a oznámí tuto skutečnost uživateli informačním hlášením, včetně důvodu odmítnutí resetu hesla, včetně uvedení kontaktu, pomocí kterého bude možno danou skutečnost reklamovat.
   b) **pokud ne** (žádný aplikační účet uživatele neodpovídá předanému ID uživatele), pak nelze reset hesla dokončit. Systém o této skutečnosti informuje uživatele informačním hlášením.
4. V případě, že existuje aplikační účet, který odpovídá předanému ID externího uživatele a byly splněny všechny kontroly v bodě 3a), pak systém vygeneruje dočasně platné

jedinečné jednorázové heslo, které bude nejméně 32 znaků dlouhé a uloží jej do atributu „dočasné_heslo“. Zároveň systém naplní atribut „platnost_dočasného_hesla“ na aktuální datum a čas plus 1 hodina. Systém nijak nemodifikuje existující heslo uživatele, stále musí existovat možnost, že se uživatel přihlásí platným heslem – ochrana proti DOS útoku zneplatněním hesla útočníkem.

5. Systém odešle e-mailovou zprávu na e-mailovou spojenou s aplikačním účtem uživatele (atribut ID_externí_uživatel), ke kterému byl proveden reset hesla. Součástí tohoto e-mailu je URL, který obsahuje informaci o ID uživatele a dále dočasné jednorázové heslo.
6. Uživatel, v případě, že chce stále obnovit heslo, klikne na URL uvedený v e-mailové zprávě nebo tento URL zkopíruje do adresního řádku prohlížeče, následně:
   a) systém ověří, zda nedošlo k vypršení platnosti jednorázového hesla (k vypršení dojde, pokud atribut „platnost_dočasného_hesla“ obsahuje hodnotu, která je nižší než aktuální systémový datum a čas v okamžiku vznesení požadavku na reset hesla). Pokud ano, oznámí to informačním hlášením uživateli a přesměruje ho za přihlašovací dialog, pokud ne, pokračuje dále,
   b) dále se pak postupuje podle Procesu změny hesla uživatele (viz kapitola 3.2 Popis procesu změny hesla uživatele) s tím rozdílem, že v tomto případě není nutné, aby uživatel zadával současně platné heslo. Po úspěšném dokončení procesu změny hesla systém nastaví:
      - atribut „dočasné_heslo = NULL“,
      - atribut „platnost_dočasného_hesla = NULL“.
7. Uživatel, v případě, že si v průběhu procesu obnovy hesla vzpomněl na původní heslo, se přihlásí platným uživatelským jménem a heslem, tj. nepoužije jednorázové heslo. V takovém případě systém při úspěšném přihlášení nastaví atributy instance objektu Uživatel takto:
      - atribut „dočasné_heslo = NULL“,
      - atribut „platnost_dočasného_hesla = NULL“.

3.3.5 Výstup procesu

Úspěšně resetované (obnovené) heslo nebo zrušení požadavku na resetování hesla standardně provedeným autentizačním procesem.

3.4 Popis procesu autentizace — externí registrovaný uživatel

3.4.1 Účel procesu

Ověřit předloženou identitu externího registrovaného uživatele a umožnit tak autorizovaný přístup k datům systému SDAT.

3.4.2 Výchozí situace

Existuje externí registrovaný uživatel, který disponuje aktivním aplikačním účtem uživatele a znalostí hesla k tomuto účtu.

3.4.3 Spouštěč procesu

Akce uživatele – uživatel chce pracovat se systémem SDAT pomocí autorizovaného přístupu a je schopen prokázat svoji identitu znalostí ID externího uživatele (e-mailové adresy) a správného hesla.

3.4.4 Průběh procesu

3.4.4.1 Hlavní úspěšný scénář

Systém ověřuje identitu externího registrovaného uživatele následujícím způsobem:

1. Systém si vyžádá zadání ID externího uživatele (e-mailové adresy) a hesla. Obě informace jsou povinné. Systém zajišťuje, že heslo není v uživatelském rozhraní zobrazeno ve viditelné podobě.
2. Uživatel zadá požadované informace a klikne na tlačítko pro přihlášení.
3. Systém zjistí, zda daný uživatel existuje, tj. pro předané ID uživatele zjistí, zda existuje právě jedna instance objektu Uživatel, kde atribut „typ uživatele = externí“ a atribut „je_registrovaný = ano“. V případě, že není nalezena právě jedna instance objektu Uživatel, která splňuje podmínky, pak systém vygeneruje hlášení „*Chybně zadané uživatelské jméno nebo heslo*“ (hlášení nesmí prozradit, zda uživatel vůbec neexistuje nebo existuje a je chybné heslo. Tato informace může pomáhat ke kompromitaci systému).
4. V případě, že existuje právě jedna instance objektu Uživatel, která splňuje výše uvedené podmínky, pak systém provede ověření dalších náležitostí:
   a) v případě, že atribut dané instance „je_aktivní = ne“, pak systém vygeneruje hlášení „*Aplikační účet uživatele <ID externího uživatele> nebyl prozatím aktivován. Přístup do systému byl odepřen.*“. Systém umožní uživateli přejít k aktivaci aplikačního účtu uživatele zadáním aktivačního klíče,
   b) v případě, že hodnota atributu „zamčený_do_kdy“ daného aplikačního účtu uživatele obsahuje nějakou hodnotu, pak systém zjistí, zda je tato hodnota (datum a čas) větší než je aktuální datum, kdy probíhá pokus o přihlášení. Pokud ano, je daný aplikační účet uživatele zamknutý a nelze se pomocí něj přihlásit. Systém vygeneruje hlášení „*Aplikační účet uživatele <ID externího uživatele> je dočasně uzamčen. Opakujte akci později. Přístup do systému byl odepřen.*“. Systém odepře přístup uživatele do systému,
   c) v případě, že je hodnota atributu „je_blokovaný = ano“, pak se jedná o blokovaný aplikační účet uživatele, pomocí kterého se nelze přihlásit. Systém vygeneruje hlášení „*Aplikační účet uživatele <ID externího uživatele> je blokován správcem systému. Přístup do systému byl odepřen.*“. Systém odepře přístup uživatele do systému,

d) zjistí, zda je uživatel přiřazen k nějakému aktivnímu, a k okamžiku přihlášení časově platnému uživatelskému místu. V případě, že ne, pak systém vygeneruje hlášení „*Aplikační účet uživatele <ID externího uživatele> není zařazen na žádné aktivní a platné uživatelské místo. Přístup do systému byl odepřen.*“. Systém odepře přístup uživatele do systému.

V případě, že probíhá přihlášení pod uživatelským účtem, který má nastaveno dočasné heslo (pro daný účet byla provedena žádost o reset hesla a proces resetování hesla nebyl dokončen, viz kapitola 3.3 Popis procesu resetování hesla uživatele. Atribut „dočasné_heslo“ obsahuje NOT NULL hodnotu), a jsou splněny všechny podmínky pro přihlášení, pak systém nastaví atributy „dočasné_heslo“ a „platnost_dočasného_hesla“ na hodnotu NULL.

5. V případě, že výše uvedené kontroly nenaleznou žádný z výše uvedených důvodů pro odepření přístupu, je identita uživatele považována za ověřenou a přístup uživateli do aplikace SDAT povolen. Systém umožňuje uživateli přístup k akcím a datům v takovém rozsahu, v jakém jsou nastavena přístupová oprávnění.

3.4.5 Výstup procesu

Úspěšně ověřená identita externího registrovaného uživatele opravňující jej k přístupu k datům systému SDAT.

3.5 Popis procesu autentizace — interní uživatel

3.5.1 Účel procesu

Ověřit předloženou identitu interního uživatele a umožnit tak autorizovaný přístup k datům systému SDAT.

3.5.2 Výchozí situace

Existuje interní uživatel, který disponuje aktivním aplikačním účtem uživatele a je zařazen do patřičných rolí interní aplikace ČNB – Řídicí databáze.

3.5.3 Spouštěč procesu

Akce uživatele – uživatel chce pracovat se systémem SDAT pomocí autorizovaného přístupu a je schopen prokázat svoji identitu pomocí standardního způsobu elektronického prokazování identity platného v ČNB.

3.5.4 Průběh procesu

3.5.4.1 Hlavní úspěšný scénář

Systém ověřuje identitu interního uživatele pomocí metody SSO (Single Sign On), kdy je uživatelské jméno získáno z operačního systému a ověřeno proti Active Directory. K SDAT mají přístup jen ti interní uživatelé, kteří jsou zařazeni v aplikační skupině „SDAT USERS“ nebo „SDAT ADMINS“ v interní aplikaci ČNB – Řídicí databáze.

Proces probíhá následovně:

1. Systém nejdříve z operačního systému klientského PC zjistí uživatelské jméno a následně ověří, zda je daný uživatel členem alespoň jedné z výše uvedených aplikačních skupin v Řídicí databázi;
   - v případě, že uživatel není členem ani jedné z výše uvedených aplikačních skupin Řídicí databáze, pak systém vygeneruje hlášení „*Aplikační účet uživatele <ID uživatele> není zařazen do žádné aplikační skupiny v Řídicí databázi. Pro zařazení uživatele do aplikačních skupin kontaktuje správce systému.*“. Přístup uživatele k systému SDAT není povolen, scénář končí.
2. Pokud je uživatel zařazen do alespoň jedné z výše uvedených aplikačních skupin Řídicí databáze, pak systém pokračuje ověřením identity uživatele proti Active Directory. Pokud toto ověření není úspěšné, systém zobrazí chybové hlášení, které získal z Active Directory. Pokud je ověření v Active Directory úspěšné, pak použije získané uživatelské jméno a zjistí, zda v rámci systému SDAT existuje právě jedna instance objektu Uživatel, kde toto uživatelské jméno odpovídá hodnotě atributu „alias_uživatele“;
   - v případě, že žádná (nebo více) instance objektu Uživatel neobsahuje v atributu „alias_uživatele“ předanou hodnotu uživatelského jména, pak systém vygeneruje hlášení „*Aplikační účet uživatele <ID uživatele> neexistuje. Pro založení aplikačního účtu uživatele a získání přístupu do aplikace se obraťte na správce systému*“. Přístup uživatele k systému SDAT není povolen, scénář končí.
3. V případě, že existuje právě jedna instance objektu Uživatel, systém provede ověření dalších náležitostí:
   a) zjistí hodnotu atributu „je_blokovaný“;
      - v případě, že hodnota tohoto atributu je nastavena na „ano“, pak se jedná o blokovaný aplikační účet uživatele, pomocí kterého se nelze přihlásit. Systém vygeneruje hlášení „*Aplikační účet uživatele <ID uživatele> je blokován správcem systému. Přístup do systému byl odepřen.*“. Přístup uživatele k systému SDAT není povolen, scénář končí,
   b) zjistí, zda je uživatel přiřazen k nějakému aktivnímu uživatelskému místu;
      - v případě, že ne, pak systém vygeneruje hlášení „*Aplikační účet uživatele <ID uživatele> není zařazen na žádné aktivní uživatelské místo. Přístup do systému byl odepřen.*“. Přístup uživatele k systému SDAT není povolen, scénář končí.
4. V případě, že výše uvedené kontroly nenaleznou žádný z výše uvedených důvodů pro odepření přístupu, je identita uživatele považována za ověřenou a přístup uživateli do aplikace SDAT je povolen. Systém umožňuje uživateli přístup k akcím a datům v takovém rozsahu, v jakém jsou nastavena jeho oprávnění.

3.5.5 Výstup procesu

Úspěšně ověřená identita interního uživatele opravňující jej k přístupu k datům systému SDAT.

3.6 Popis procesu přístupu neregistrovaného uživatele

3.6.1 Účel procesu

Hlavním účelem procesu je umožnit externím uživatelům ve vybraných případech dodat data do ČNB bez nutnosti se registrovat v systému SDAT. Tento proces se bude používat tehdy, pokud nastala nějaká skutečnost nebo událost, která není předem známa ČNB a ČNB tak nemůže předepsat vykazovací povinnost k určitému datu a určité Osobě. Dochází tu tak k obrácení celého standardního procesu plánování vykazovacích povinností. Zatímco běžně funguje vykazování tak, že ČNB předepíše vykazovací povinnost a Osoba ji splní (zašle data s referencí na předepsanou vykazovací povinnost), zde vykazovací povinnost sice existuje, ale není známo, kdy a kým bude plněna.

3.6.2 Výchozí situace

S ohledem na skutečnost, že sběr dat od neregistrovaných uživatelů bude představovat pouze jednotky procent z celkového objemu sbíraných dat, a jedná se tak vlastně o výjimku ze standardního procesu, je třeba řešení koncipovat tak, aby v maximální možné míře byla zachována pravidla platná pro běžný proces sběru dat, kdy ČNB předepisuje Vykazovací povinnosti a Osoby je plní. Jedná se především o tato základní pravidla:

- každý uživatel vstupující do systému SDAT, který má do systému zapisovat údaje, tak činí prostřednictvím aplikačního účtu uživatele,
- každému aplikačnímu účtu uživatele jsou přidělena přístupová práva prostřednictvím zařazení uživatele na uživatelské místo,
- data (tzv. Vydání výskytu výkazu) nelze přijmout, aniž by byla definována Vykazovací povinnost (ke každému Vydání výskytu výkazu musí existovat právě jedna instance objektu Výskyt výkazu).

Je požadováno, aby přístup neregistrovaných uživatelů byl možný pouze přes webovou aplikaci systému SDAT a vyžadoval pouze nezbytné minimum identifikačních údajů. Neregistrovaní uživatelé nemohou využít webovou službu ani jiný komunikační kanál. Důvodem je to, že je třeba neregistrovanému uživateli nabídnout určitý komfort při tvorbě hlášení. To nelze webovými službami, případně jinými komunikačními kanály, dosáhnout. Tímto způsobem bude možné zasílat pouze omezený okruh jednoduchých výkazů, ke kterým však mohou být připojeny i binární soubory.

3.6.3 Spouštěč procesu

Osobě vznikne povinnost zaslat hlášení do ČNB a uživatel Osoby klikne ve veřejně přístupné zóně (Veřejná část webové aplikace SDAT) na tlačítko VSTUP BEZ REGISTRACE.

3.6.4 Průběh procesu

Hlavní úspěšný scénář

Proces přístupu neregistrovaného uživatele probíhá následovně:

1. Systém zobrazí uživateli formulář, kde si vyžádá zadání křestního jména, příjmení a e-mailové adresy uživatele. Dále si systém vyžádá zadání kontrolního, náhodně vygenerovaného, řetězce, pomocí kterého bude následně proveden CAPTCHA test. CAPTCHA test je do aplikace integrován tak, aby byl pro koncového uživatele co nejméně náročný (například tzv. „reCaptcha“), případně umožňuje uživateli získat nový kód (pokud je stávající kód uživateli nečitelný) a umožní získat kód pomocí přehráním zvuku.
2. Uživatel vyplní formulář a odešle data ke zpracování. Systém provede v této fázi zpracování kontrolu – CAPTCHA test. V případě, že CAPTCHA test proběhl v pořádku, pokračuje scénář dalším bodem, v opačném případě vygeneruje nový CAPTCHA kód a scénář se vrací do bodu 1. (již jednou vyplněné údaje zůstávají vyplněné).
3. Systém bez součinnosti s uživatelem provede následující akce:
   a) vytvoří uživatele (aplikační účet uživatele):
      i) jako ID uživatele bude použita zadaná e-mailová adresa. V tomto případě systém nevykoná kontrolu na duplicitní ID (viz kapitola 2.2 Objekt Uživatel),
      ii) označí daný aplikační účet uživatele atributem „je registrovaný = ne“ a „je_aktivní = ano“,
      iii) s ohledem na fakt, že se jedná o zřízení jednorázového přístupu, systém nebude od uživatele požadovat zadání hesla, ani jej nebude automaticky generovat. Nebude vyžadováno ani dodatečné ověření identity pomocí dvoukrokové autentizace,
   b) vytvoří novou Osobu v Registru osob, jako název použije jméno a příjmení zadané v bodě 1. Osobu je nutno vytvořit proto, aby bylo možno zachovat celý model vykazovacích povinností, tedy kaskádu Vykazovací povinnost - Výskyt výkazu – Vydání výskytu výkazu, která bez existence Osoby nemůže existovat. Systém zařadí Osobu do Typu osoby „Neregistrované Osoby“. Součástí tohoto typu Osoby jsou nadefinovány vykazovací povinnosti. Tím vlastně neregistrovaná osoba získá seznam Výkazů, za které může (jako neregistrovaná) vykazovat (dodávat data),
   c) vytvoří uživatelské místo typu „externí“ s atributem „je_registrovanné“ nastaveným na hodnotu „ne“, toto uživatelské místo spojí s Osobou vytvořenou v bodě b) a naváže na ně takové role a aktivity, které jsou určeny pro vykazování dat (vzorové uživatelské místo, TYP1),
   d) zařadí uživatele na uživatelské místo vytvořené v kroku c).
4. Systém nabídne neregistrovanému uživateli seznam Výkazů, které jsou určeny pro tento typ uživatelů (seznam Výkazů vznikne na základě zařazení Osoby do Typu osoby).
5. Uživatel vybere výkaz, který chce zaslat ČNB.
6. Systém na základě znalosti o tom, o jaký Výkaz se jedná (viz bod 5.) dohledá patřičnou instanci objektu Vykazovací povinnost a z ní odvodí a následně vytvoří instanci objektu Výskyt výkazu. Výskyt výkazu naváže k Osobě (viz bod 3.b).
7. Systém umožní uživateli Výkaz vyplnit. Po dokončení vyplnění dat uživatel připraví výkaz k odeslání.
8. Uživatel odešle data za vybraný Výkaz ke zpracování.
9. Systém vytvoří Vstupní zprávu, do které vloží předložené vydání výskytu výkazu a odešle ji ke zpracování.

10. Systém zahájí proces Zpracování Vstupní zprávy. Po jeho dokončení je podána zpráva uživateli o jeho výsledku. Systém uživateli nabídne možnost odeslání dalšího Výkazu. V takovém případě se pokračuje od bodu 4. Pokud uživatel nepotřebuje odesílat další Výkaz, systém ukončí tzv. session a pokud kdykoli v budoucnu vznikne potřeba odeslat další Výkaz, postupuje se od bodu 1.

3.6.5 Výstup procesu

Zpracovaná data od neregistrované osoby.

3.7 Popis procesu vytvoření externí Osoby a aplikačního účtu uživatele externím subjektem (Autoregistrace)

3.7.1 Účel procesu

Základním účelem procesu autoregistrace je umožnit externímu uživateli, aby se ve vybraných případech sám zaregistroval v systému SDAT tak, aby nadále mohl pravidelně plnit svou vykazovací povinnost. Tímto procesem dojde k vytvoření nové Osoby v systému SDAT, zřízení alespoň jednoho aplikačního účtu uživatele a tomuto účtu prostřednictvím uživatelského místa přiřazení přístupových práv (rolí), která uživateli umožní přihlášení se k SDAT za účelem odeslání předepsaného Výkazu(ů) a určení alespoň jednoho výkazu, který bude Osoba odesílat. Celý tento proces je třeba vykonat bez nutnosti jakékoli akce na straně ČNB.

3.7.2 Výchozí situace

Výchozí situace je následující:

- existuje Osoba (resp. uživatel, který ji zastupuje), která sama rozpozná, že jí vzniká vykazovací povinnost, tzn. předem neznámá osoba pro SDAT. Toto zjištění může pramenit například z veřejně přístupných stránek ČNB, kde zároveň existuje odkaz na registrační formulář, který umožní Osobu registrovat anebo je taková Osoba oslovena ČNB se žádostí zapojit se do dobrovolného procesu sběru dat,
- tato předem neznámá Osoba prozatím není evidována v Registru osob a nemá aplikační účet uživatele v aplikaci SDAT,
- vytvoření přístupu do SDAT pomocí autoregistrace NESMÍ VYŽADOVAT žádnou akci na straně ČNB (zjednodušení nároků na administraci systému),
- mohou existovat subjekty, které použijí autoregistraci (budou to právnické osoby a budou mimo ČR), které nebudou mít přiděleno české IČO, ale nějaký svůj národní identifikátor,
- subjekty vzniklé autoregistrací dostanou přiřazen jen úzký okruh Výkazů (je předem znám),
- během procesu autoregistrace **nebude vyžadováno zadání žádné informace o kvalifikovaném certifikátu uživatele**. Toto omezení je definováno na základě znalosti toho, jaké výkazy potřebují předem neznámé osoby vykazovat. Jedná se o výkazy, u nichž nebude podpis kvalifikovaným certifikátem nutný. V případě, že někdy později vznikne

autoregistrovanému uživateli potřeba zaslat výkaz, u kterého je podpis kvalifikovaným certifikátem nutný, bude existovat možnost, po přihlášení se k systému, k uživatelskému účtu uložit informace o certifikátu určeném pro vytváření elektronického podpisu a následně elektronickým podpisem (kvalifikovaným certifikátem) daný výkaz podepsat,

- v rámci autoregistrace je třeba vykonat kontroly na to, zda Osoba již existuje v SDAT. Analýzou bylo zjištěno, že má smysl rozdělit subjekty na tři základní kategorie. Důvodem pro členění do tří různých kategorií je to, že každá z kategorií disponuje jinou možností, jak Osobu identifikovat. Byly určeny tyto kategorie:
  - **fyzická osoba – nepodnikatel.** Za duplicitní Osobu bude považována ta osoba, která se bude shodovat s již existující osobou v systému SDAT v těchto čtyřech atributech – „jméno_uživatele“, „příjmení_uživatele“, „datum_narození“, „email“,
  - **právnická osoba (včetně fyzické osoby podnikatele) se sídlem na území ČR.** Za duplicitní osobu bude považována ta osoba, která se bude shodovat s již existující osobou v systému SDAT v **atributu IČO** nebo v kombinaci atributů „název“ a „právní_forma“,
  - **právnická osoba se sídlem mimo území ČR**. Za duplicitní osobu bude považována ta osoba, která se bude shodovat s již existující osobou v systému SDAT v kombinaci atributů „kód_země“ a „národní_identifikátor“.

3.7.3 Spouštěč procesu

Osobě vznikne povinnost zaslat hlášení do ČNB nebo se dobrovolně rozhodne účastnit se nějakého statistického šetření. Předem neznámá osoba klikne ve veřejně přístupné zóně (Internetové stránky ČNB, veřejně přístupná část aplikace SDAT) na tlačítko REGISTROVAT NOVOU OSOBU.

3.7.4 Průběh procesu

3.7.4.1 Hlavní úspěšný scénář

Proces popisuje vytvoření nové Osoby, její označení za Vykazující osobu a vytvoření aplikačního účtu uživatele, pomocí kterého bude možno vykonávat aktivity v systému SDAT (vykazovat data) probíhá následovně:

1. Systém zobrazí registrační formulář sloužící pro autoregistraci předem neznámého subjektu do aplikace SDAT.
2. Uživatel vybere jednu z následujících možností (určí tzv. Kategorii osoby):
   a) fyzická osoba – nepodnikatel,
   b) právnická osoba – ČR,
   c) právnická osoba – mimo ČR.
3. Systém nabídne uživateli na základě výběru Kategorie osoby v bodě 2. atributy, které je uživatel povinen vyplnit. Rozsah atributů, jejich povinnost a způsob jejich validace je určen nastavením systému (viz dokument C – Vykazovací povinnosti a Registr osob).
4. Uživatel vyplní atributy o Osobě a prokáže, že není stroj (CAPTCHA test; ochrana proti DOS/DDOS útoku). Systém provádí kontrolu na existenci osoby, tj. na základě zadaných údajů v bodě 3. provede kontrolu na duplicitu. Budou aplikovány tyto kontroly:

a) **fyzická osoba** – pokud se bude shodovat zadaná kombinace jméno/příjmení/datum nebo rok narození/e-mailová adresa s nějakým existujícím subjektem v SDAT, bude prohlášeno zadání za duplicitní a systém neumožní takovou osobu vytvořit. Místo toho nabídne uživateli se pomocí zadaného e-mailu přihlásit, případně umožní na zadanou e-mailovou adresu odeslat e-mail, který umožní obnovit zapomenuté heslo,
b) **právnická osoba – ČR** – pokud se bude shodovat zadaná hodnota **v atributu „IČO“** s nějakým existujícím subjektem v SDAT, bude prohlášeno zadání za duplicitní a systém neumožní takovou osobu vytvořit,
c) **právnická osoba – ČR** – pokud se bude shodovat zadaná kombinace název/právní forma s nějakým existujícím subjektem v SDAT, bude prohlášeno zadání za duplicitní a systém neumožní takovou osobu vytvořit,
d) **právnická osoba – mimo ČR** – pokud se bude shodovat zadaná kombinace kód země/národní identifikátor s nějakým existujícím subjektem v SDAT, bude prohlášeno zadání za duplicitní a systém neumožní takovou osobu vytvořit.

5. Zakládaná Osoba bude vždy vedena jako Vykazující osoba. Uživatel může určit, že má k této osobě vztah jako Zastupující osoba. V takovém případě systém umožní nadefinovat, v jakém rozsahu Vykazující osobu zastupuje a umožní přiřadit a pokud neexistuje, tak i vytvořit, druhou osobu, která bude do systému zavedena jako Zastupující osoba. Pokud uživatel určí, že je třeba vytvořit k Vykazující osobě ještě Zastupující osobu, pak systém vykoná následující:
   a) pokud uživatel v bodě 4. určí, že je třeba vytvořit k Vykazující osobě Zastupující osobu, pak systém nabídne uživateli možnost definovat, v jakém rozsahu bude zastupování prováděno (jedná se o určení Výkazů, které má zastupující osoba za Vykazující osobu dodávat),
   b) systém nabídne seznam všech Výkazů, které jsou určeny pro autoregistrované osoby,
   c) uživatel vybere alespoň jeden Výkaz. Vzhledem k tomu, že definice zastupování znamená i vytvoření rozsahu platnosti tohoto zastupování, bude (pokud není uvedeno v dalším textu jinak) jako platnost_od použito aktuální systémové datum a jako platnost_do bude použito maximální datum (31. 12. 4000),
   d) systém si vyžádá informaci o Zastupující osobě. Zastupující osobou může být jakákoli fyzická nebo právnická osoba, která má sídlo na území ČR. Systém tedy při definici zastupující osoby postupuje stejně jako v bodech 2. až 4. s tou výjimkou, že jako kategorii Osoby nenabízí „Právnická osoba – mimo ČR“. Jsou aplikovány stejné kontroly jako v případě bodu 4.,
   e) v případě, že uživatel splnil všechny omezující podmínky zapsané výše (vybral alespoň jeden Výkaz a zadal neduplicitní Osobu), pak systém vytvoří novou Osobu a zařadí ji do role „Zastupující osoba“ s rozsahem platnosti od definovaného v bodě c), pro Výkazy vybrané uživatelem v bodě b) a pro Vykazující osobu vytvořenou uživatelem během hlavního úspěšného scénáře. Kontrola na duplicitu zastupování není v tuto chvíli potřeba, protože definice zastupování probíhá k právě nově vytvořené Vykazující osobě, a tudíž ta žádné jiné zastupování ještě vytvořeno nemá.
6. Systém vytvoří novou osobu v Registru osob ve stavu „neverifikovaná osoba“. Zároveň tuto osobu zařadí do role „Vykazující osoba“. Systém přidělí dané Osobě unikátní ID (navíc k systémovému ID), které sdělí Osobě.
7. V případě, že nebylo požadováno vytvoření Zastupující osoby, systém umožní uživateli určit, jaké Výkazy chce do ČNB zaslat. Systém nabídne seznam všech Výkazů, které jsou určeny pro autoregistrované osoby.

8. Systém zobrazí formulář pro vytvoření aplikačního účtu uživatele. Pro založení aplikačního účtu uživatele je třeba zadat:
   a) jméno uživatele,
   b) příjmení uživatele,
   c) e-mailová adresa uživatele,
   d) telefonní číslo uživatele v mezinárodním formátu[6], na kterém je možno přijímat SMS zprávy,
   e) dvakrát shodně heslo, které odpovídá bezpečnostní politice aplikace SDAT.
9. Uživatel vyplní formulář pro vytvoření aplikačního účtu uživatele a prokáže, že není stroj (CAPTCHA test).
10. Systém provede kontrolu na validitu a duplicitu zadaných dat. Pokud kontrola proběhne v pořádku, pak **systém provede vytvoření aplikačního účtu uživatele** v rozsahu dat zadaných v bodě 8..
11. Systém provede dodatečné ověření identity uživatele (viz UMU_14.0), a pokud je vše v pořádku, pak systém nastaví aplikační účet uživatele jako aktivní.
12. Systém vytvoří tzv. hlavní uživatelské místo, kterému budou přiřazena taková oprávnění, aby bylo možno provádět vykazování a zakládání dalších aplikačních účtů v rámci Osoby. Toto uživatelské místo bude navázáno:
   a) k Vykazující osobě v případě, že nebyla využita možnost vytvoření Zastupující osoby,
   b) k Zastupující osobě v případě, že byla využita možnosti vytvoření Zastupující osoby. Vykazující osoba v systému tak existuje BEZ jakéhokoli aplikačního účtu uživatele.
13. Systém automaticky zařadí Vykazující Osobu do Typu osoby, který je určen pro „předem neznámé osoby“.
14. Proces autoregistrace končí.

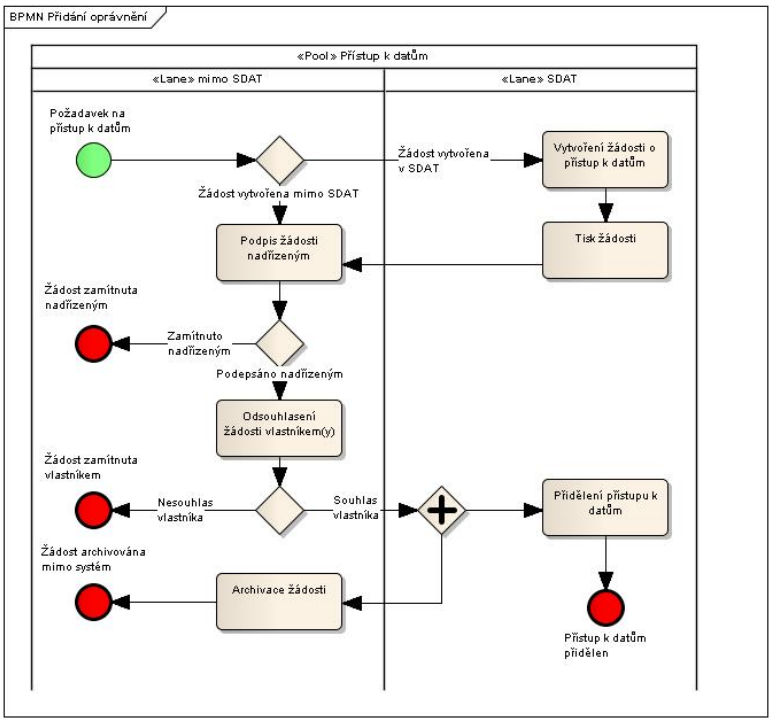
3.7.5 Výstup procesu

Úspěšně zaregistrovaná Vykazující osoba (případně i Zastupující osoba) v SDAT zařazená do typu „předem neznámé osoby“, vytvořený, ověřený a aktivní aplikační účet uživatele.

3.8 Proces Přidělení oprávnění pro přístup k datům

Cílem procesu je zajištění přístupu k datům (Hodnotám údajů) pro oprávněné interní uživatele. Tento přístup je udělován na základě žádosti interního uživatele (viz kapitola 3.8.1 Subproces vytvoření žádosti o přístup k datům, která je dále schválena příslušnými vedoucími pracovníky mimo systém SDAT. Na základě podepsané žádosti provede administrátor systému patřičné nastavení oprávnění (viz kapitola 3.8.2 Subproces definování přístupu k datům.

[6] Viz E.123: Notation for national and international telephone numbers, e-mail addresses and web addresses - http://www.itu.int/rec/T-REC-E.123-200102-I/e



Obrázek 2 - Proces přidělení oprávnění přístupu k datům

Systém SDAT:

- poskytuje podporu procesu vytvoření žádosti a přidělení přístupu k povoleným datům,
- neposkytuje podporu:
  - podepisování dokumentu funkcemi SDAT; žádost je procesně zpracovávaná mimo systém SDAT,
  - vytvoření žádosti pro zaměstnance věcné a technické správy, kteří mají neomezený přístup, a zaměstnance, žádající o přístup ke všem Výkazům s necitlivými daty pro vyjmenované Vykazující osoby; vlastní přidělení přístupu k datům těchto zaměstnanců systém SDAT podporuje.

3.8.1 Subproces vytvoření žádosti o přístup k datům

3.8.1.1 Účel procesu

Základním účelem procesu je vytvořit v ČNB pro zaměstnance žádost o přidělení přístupu k datům (Hodnotám údajů). Funkcionalita je přístupná všem zaměstnancům ČNB. Zaměstnanci nemusí mít vytvořen aplikační účet v SDAT.

3.8.1.2 Výchozí situace

O přístup k datům žádá (dále jen „žadatel“):

- zaměstnanec (pro sebe, nebo — v případě požadavku na shodný přístup — pro více zaměstnanců ze shodného organizačního útvaru),
- administrátor pro kteréhokoliv zaměstnance nebo skupinu zaměstnanců ze shodného útvaru.

Žádost zaměstnance/ců, pro kterého/které se žádá o přístup k datům, je třeba podepsat nadřízeným ředitelem (ředitel odboru/samostatného odboru/sekce) daného zaměstnance, který o přístup žádá:

- žádost o citlivá data podepisuje ředitel sekce/samostatného odboru,
- žádost o necitlivá data podepisuje ředitel odboru/samostatného odboru.

Při schvalování, které probíhá mimo systém SDAT, se postupuje následovně:

- k přístupu k citlivým datům dává souhlas ředitel sekce/samostatného odboru, do které spadá odbor vlastníka dat,
- k přístupu k necitlivým datům dává souhlas ředitel odboru/samostatného odboru vlastníka dat s výjimkou zaměstnanců vybraných útvarů ČNB, kde není souhlas ředitele odboru/samostatného odboru, který je vlastníkem Výkazu, požadován. Tyto vybrané útvary systém eviduje a umožňuje administrátorovi je aktualizovat.

Žádost zaměstnance/ců, pro kterého/které se žádá o přístup, je třeba vždy podepsat všemi příslušnými řediteli.

3.8.1.3 Spouštěč procesu

Proces je spouštěn ad-hoc žadatelem, pokud v rámci pracovní náplně zaměstnance (nebo pracovní náplně skupiny zaměstnanců ze shodného organizačního útvaru) je potřeba mít přístup k datům, ke kterým zatím nemá v SDAT definován přístup.

3.8.1.4 Průběh procesu

Žádosti o přístup k datům žadatel vytváří prostřednictvím průvodce.

Proces tvorby žádosti o oprávnění k datům probíhá následovně:

1. Žadatel (dále pouze „uživatel“) spustí průvodce pro vytvoření žádosti. Možnost spustit průvodce je dostupná i pro uživatele, kteří nejsou uživateli systému SDAT.
2. Systém nabídne uživateli seznam útvarů ČNB, který získá v reálném čase z Řídicí databáze. Uživatel vybere právě jeden útvar.
3. Systém uživateli nabídne seznam zaměstnanců daného útvaru vybraného v bodě 2., který také získá v reálném čase z Řídicí databáze.
4. Uživatel určí, zda předmětem žádosti je oprávnění k citlivým nebo necitlivým údajům:
   a) v případě žádosti o oprávnění k **necitlivým údajům** systém umožní uživateli definovat požadavek pomocí následujících kombinací ze seznamu:

i) výkazů[7] (nabídka Vykazovací rámec nebo konkrétní Výkaz),
ii) vykazujících osob (nabídka Typ osoby nebo konkrétní Vykazující osoba nebo „vše“),
iii) pokud je použit dynamický přístup (definice části oprávnění přes Typ osoby, Vykazovací rámec), je možno zadat ke každému rozsahu oprávnění negativní výjimku (označením Výkazu nebo Vykazující osoby ze seznamu Výkazů nebo Vykazujících osob, zahrnutých do dynamické definice odpovídajícího rozsahu oprávnění. Tyto označené Výkazy nebo Vykazující osoby nebudou zahrnuty do rozsahu oprávnění),

b) v případě žádosti o oprávnění k **citlivým údajům** systém umožní uživateli definovat požadavek pomocí následujících kombinací ze seznamu:
i) výkazů s citlivými daty (konkrétní Výkaz)[8],
ii) vykazujících osob (konkrétní Vykazující osoba, Typ osoby, „vše“),
iii) pokud je použit dynamický přístup přes Typ osoby, je možno zadat negativní výjimku (označením Vykazující osoby ze seznamu Vykazujících osob zahrnutých do dynamické definice. Tyto označené Vykazující osoby nejsou zahrnuty do rozsahu oprávnění).

5. Uživatel opakuje definici dle 4.a) (v případě necitlivých údajů) nebo 4.b) (v případě citlivých údajů) tolikrát, kolik různých kombinací oprávnění Výkaz/Osoba požaduje přidělit. Po dokončení této definice pokračuje proces dalším bodem.
6. Uživatel vyplní textové pole „důvod žádosti“. Jedná se o textové pole velkého rozsahu, tak aby uživatel nebyl limitován počtem znaků pro zdůvodnění své žádosti.
7. Systém provede rekapitulaci žádosti jejím výpisem na obrazovku s možností:
   a) vrátit se zpět ke každému z již proběhlých kroků průvodce s možností změnit jakoukoli z dříve zadaných hodnot,
   b) dokončit proces tvorby žádosti a vygenerovat vlastní žádost.
8. V případě, že uživatel dokončí proces tvorby žádosti, pak systém vytvoří vlastní žádost ve formátu PDF takto:
   a) zobrazí identifikaci zaměstnance, který žádost vytvořil (osobní číslo, jméno a příjmení, funkci, organizační zařazení),
   b) zobrazí identifikaci zaměstnance(ů), pro které je žádost vytvořena (ve stejném rozsahu jako je uvedeno v bodě a),
   c) zobrazí
      i) pro výkazy bez citlivých dat identifikaci:
         - nadřízeného ředitele odboru/samostatného odboru, který je vlastníkem dat a místo na podpis,
         - ředitele odboru/samostatného odboru, který je vlastníkem dat a místo na podpis,
      ii) pro výkazy s citlivými daty identifikaci:

[7] Nabídka obsahuje Výkazy bez citlivých dat a s citlivými daty, přístup však je umožněn jen k Datovým oblastem bez citlivých dat.

[8] U žádosti o přístup k citlivým údajům nelze použít dynamickou definici na straně Výkazu, tedy nelze aplikovat definici přes instance objektu Vykazovací rámec.

- nadřízeného ředitele odboru/samostatného odboru žadatele o přístup a místo na podpis,
- nadřízeného ředitele sekce žadatele o přístup a místo na podpis,
- ředitele sekce, do jehož působnosti spadá vlastník dat a místo na podpis,

d) algoritmus určení vlastníka dat je následující:
   i) pro každý Výkaz, pro jehož data je žádán přístup, se načte vlastník dat z atributu „vlastník_dat“ objektu Výkaz. V tomto atributu je vždy uvedena identifikace útvaru vlastníka dat, nikoli jméno konkrétního ředitele. Vzniká tak určení tzv. „defaultního vlastníka dat“,
   ii) následně je pro každý Výkaz zjištěno, zda je vlastnictví dat rozděleno mezi více útvarů. Rozdělit vlastnictví mezi více útvarů je možno při tzv. sektorovém přístupu k vlastnictví dat, kdy existuje jeden výkaz, který je vykazován Osobami z více různých sektorů (například bankami a pojišťovnami). Sektory jsou v systému SDAT reprezentovány pomocí objektu Typ osob. Z hlediska systému SDAT se jedná o zjištění, zda pro daný Výkaz existuje nějaká, k aktuálnímu okamžiku platná, instance objektu Určení vlastníka dat. V případě, že ano, pak systém načte všechny takové instance a z atributu „vlastník_dat“ objektu Určení vlastníka dat zjistí všechny nadefinované vlastníky dat pro různé Typy osob,
   iii) následně systém zjistí, pro jaké Osoby je (pro zkoumaný Výkaz) požadováno přidělení oprávnění,
      - pokud pro daný výkaz neexistuje žádná definice vlastníka dle typu osoby, pak bude vyžadován pouze podpis tzv. defaultního vlastníka dat. Tento jediný podpis bude znamenat schválení přístupu k datům daného výkazu za všechny Osoby, které jsou v žádosti (v kombinaci se zkoumaným výkazem) uvedeny,
      - pokud pro daný Výkaz existuje nějaká definice vlastníka dle typu osoby, pak bude vyžadováno tolik podpisů různých ředitelů, kolik různých Osob z různých typů osob se v žádosti vyskytuje. V případě, že je v žádosti uvedena osoba, která je zařazena v typu osoby, pro kterou není definován žádný vlastník dat, bude jako vlastník dat použit tzv. „defaultní vlastník dat“, viz předchozí bod,

e) systém provede vyhodnocení vlastníka dat za každý Výkaz, ke kterému je požadován přístup. Na výsledné žádosti se musí objevit identifikace a prostor pro podpis všech ředitelů, kteří jsou vlastníky alespoň jedné kombinace Osoba/Výkaz, která je na žádosti uvedena,
f) systém zobrazí přehled kombinací Výkaz/Osoba pro které je požadováno udělení oprávnění,
g) systém zobrazí prázdné pole pro zapsání referenčního čísla (referenční číslo bude vygenerováno v návazném procesu zpracování žádosti mimo systém SDAT).

9. Systém nabídne uživateli možnosti uložit a vytisknout vygenerovanou žádost.
10. Uživatel žádost uloží (mimo SDAT) a proces vytvoření žádosti končí.

3.8.1.5 Výstup procesu

Výstupem procesu je:

- v systému SDAT: přidělení přístupu k datům žadateli/uživateli,

- mimo systém SDAT: vygenerovaná a archivovaná žádost o přístup k datům ve formátu PDF.

3.8.2 Subproces definování přístupu k datům

3.8.2.1 Účel procesu

Základním účelem procesu je definovat pro uživatele v ČNB přístup k datům (Hodnotám údajů). Funkcionalita je přístupná uživateli na UM určeném pro správu uživatelských oprávnění systému SDAT (administrátor).

3.8.2.2 Výchozí situace

Přístup k datům zakládá a aktualizuje administrátor na základě žádosti podepsané příslušnými vedoucími zaměstnanci. Žádost je zpravidla vygenerována systémem SDAT nebo pro atypické případy vzniklá jiným způsobem mimo SDAT.

3.8.2.3 Spouštěč procesu

Proces je spouštěn ad-hoc administrátorem, kterému zaměstnanec ČNB doručí žádost o přístup k datům nebo požadavek na odebrání přístupu k datům.

3.8.2.4 Průběh procesu

Administrátor přístup k datům aktualizuje prostřednictvím průvodce podle požadavků v žádosti o přístup nebo odebrání přístupu k datům, které obdržel mimo SDAT.

1. Přidělování přístupu k datům

Rozsah oprávnění administrátor definuje staticky nebo dynamicky, vybírá požadované hodnoty z nabídky z listu (grid) zvlášť za Výkazy bez citlivých dat a zvlášť za Výkazy s citlivými daty. Pokud zadá nepovolené kombinace (viz kapitola 2.9 Objekt Rozsah oprávnění), systém na to upozorní a nepustí jej dál. K jednotlivým rozsahům oprávnění definovaným dynamicky administrátor může zadat výjimku (označením nabídky z listu Výkazů nebo Vykazujících osob, zahrnutých do dynamické definice).

2. Ukončení přístupu k datům

Administrátor ukončuje platnost přístupu uživatelů k datům (označením jednoho uživatele nebo hromadně) tak, že zobrazí stávající oprávnění a podle povahy věci buď ukončí platnost všem definovaným rozsahům oprávnění anebo jen požadovaným. Systém zároveň umožňuje změnit přístup k datům ukončením platnosti výjimky.

3.8.2.5 *Výstup procesu*

Výstupem procesu je aktualizovaný přístup uživatelů k datům.

4 Bezpečnostní politika

V rámci aplikace SDAT se předpokládá přístup dvou základních typů uživatelů (aplikačních účtů), tj. interních uživatelů (zaměstnanci ČNB) a externích uživatelů (zaměstnanci Osob). Toto členění má zásadní vliv z hlediska zavedení politiky hesel. Bude platit:

- **interní uživatel** - zaveden v Active Directory ČNB, stejně tak jako je zařazen v Řídicí databázi. Jako takový nemusí mít v systému registrováno žádné heslo, protože ověření jeho identity proběhne pomocí SSO. I když se počítá s tím, že primárním identifikátorem každého aplikačního účtu uživatele je e-mailová adresa, interní uživatel bude vybaven navíc tzv. aliasem aplikačního účtu uživatele, ve kterém bude uchovávána informace o jeho interním ID, pomocí kterého je v ČNB vykonávána SSO autentifikace (U0****),
- **externí uživatel** – přistupuje k systému z prostředí Internetu a není možné jeho zavedení do Active Directory ani do Řídicí databáze. Pro ověření identity takového uživatele je zvolena metoda ověření přes zadání uživatelského jména a hesla[9]. Tento princip bude v některých případech (například změna hesla u uživatelů vzniklých autoregistrací) zesílen o další autentizační kanál, kterým bude zasílání jednorázového hesla (PINu) na mobilní telefon. Externího uživatele dále dělíme na registrovaného a neregistrovaného (viz kapitola 2.2 Objekt Uživatel a atribut „je registrovaný“). Z bezpečnostní politiky jsou vyňati uživatelé, kteří přistupují v rámci tzv. neregistrovaného přístupu. Tito uživatelé nepodléhají autentifikaci a nemají žádné heslo.

S ohledem na fakt, že pro správu hesel externích uživatelů není možné využít nějaký existující prostředek, který určuje politiku hesel, je třeba pro systém SDAT nadefinovat vlastní aparát, který tuto definici zařídí.

4.1 Systém uložení hesel

Veškerá hesla pro všechny externí (registrované) aplikační účty budou uložena v aplikaci SDAT. Aplikační logika zajistí, že heslo nebude uloženo v otevřeném tvaru, ale naopak bude šifrováno pomocí kryptografické funkce navržené pro šifrování hesel (například Bcrypt nebo SCrypt nebo jiné; není povoleno použít algoritmy MD5 a SHA-1). Zašifrované heslo výše uvedeným postupem dále nazýváme „otisk hesla“

Dalším bezpečnostním prvkem, který bude použit pro vytvoření otisku hesla, je metoda tzv. **solení hesla (password salt)**. Účelem této metody je zajistit, aby dvě stejná hesla neměla v databázi stejný otisk hesla (hash). Toto bude zajištěno tak, že před vytvořením otisku (hashe) hesla bude k heslu, které zadá uživatel, přidán unikátní a předem neodhadnutelný

[9] Metoda ověření přes kvalifikovaný certifikát byla vyloučena – jednak dle výkladu zákona není možné kvalifikovaný certifikát využít k ověřování identity, a také je třeba předpokládat, že budou existovat uživatelé, u kterých kvalifikovaný certifikát nebude povinný (autoregistrace).

řetězec. Řetězec, který bude použit jako sůl, bude uložen v databázi v otevřené podobě (viz objekt Heslo uživatele, atribut sůl).

Hesla jsou v databázi ukládána vždy k aplikačnímu účtu uživatele, ke kterému se váží (kardinalita 1:N; jeden aplikační účet uživatele smí mít neomezeně hesel, každé heslo se váže právě k jednomu aplikačnímu účtu uživatele)[10]. Kromě samotného otisku hesla jsou evidovány následující atributy:

- **„platnost_od“** – datum a čas od kdy dané heslo platí,
- **„platnost_do“** – datum a čas do kdy dané heslo platí. Tento údaj je třeba chápat jako údaj o maximálním datu a času, do kdy je heslo platné. Po jeho uplynutí je heslo exspirované. Uživatel je schopen toto datum změnit tak, že si změní heslo ještě před uplynutím data exspirace,
- **„je_platné“ –** boolean (true/false) udávající, zda je dané heslo platné nebo ne. Platné smí být vždy právě jedno heslo. Tento atribut je zaveden proto, aby bylo možno prokazatelně identifikovat heslo, proti kterému bude prováděn proces ověření identity uživatele. V tomto případě se nelze spolehnout na datum platnosti od/do, protože heslo již mohlo exspirovat. Samotná exspirace hesla však není důvodem k jeho zneplatnění.
- **„sůl“ –** řetězec, který je potřeba přidat k heslu uživatele před tím, než bude vytvořen otisk hesla.

Systém musí zajistit, aby k předanému uživatelskému účtu bylo vždy možno zjistit právě jedno platné heslo – to znamená, že v jeden časový okamžik smí být platné maximálně jedno heslo. Situace, kdy v daný okamžik není platné žádné heslo, je také validní (například interní uživatelé a externí (neregistrovaní) uživatelé nebudou mít vůbec žádné heslo, anebo heslo vypršelo).

4.2 Popis procesu ověření identity uživatele pomocí hesla

Následující text popisuje algoritmus ověření identity uživatele pomocí uživatelského jména a hesla (algoritmus je platný pouze pro přihlášení externího registrovaného uživatele). Níže uvedený algoritmus je pouze rámcový a nelze jej chápat jako detailní UseCase procesu přihlašování (nepracuje s konfiguračními položkami, které přihlašovací proces ovlivňují).

Proces probíhá následovně:

1. **Uživatel zadá uživatelské jméno a heslo** v rámci přihlašovacího dialogu.
2. **Systém zjistí, zda uživatelské jméno existuje v systému:**
   - pokud ano, pokračuje se bodem 3.,
   - pokud uživatelské jméno v systému neexistuje, vrací systém hlášení „Neplatné uživatelské jméno nebo heslo“[11] a scénář končí.

[10] Platí pouze pro aplikační účty externích uživatelů, interní uživatelské účty nebudou mít žádné heslo.

[11] Zcela záměrně bude hlášení vždy odkazovat na neplatnost uživatelského jména nebo hesla, i když systém bude vědět, že chyba je v uživatelském jméně. Pokud by systém vrátil hlášku zcela přesně, mohlo by to útočníkovi poskytnout informaci o tom, jaká uživatelská jména v systému jsou, a usnadnilo by mu to kompromitaci systému.

3. Systém z databáze k předanému uživatelskému účtu **zjistí platný otisk hesla** (související instance objektu Heslo uživatele, atribut „je_platné = ano“).
4. **Systém provede vytvoření otisku hesla** z údajů zadaných uživatelem v rámci přihlašovacího dialogu a to takto:
   a) systém vytvoří řetězec, který vznikne spojením tzv. soli (viz výše) a zadaného hesla,
   b) systém vytvoří hash daného řetězce pomocí stejného algoritmu, jakým byl vytvořen otisk hesla v databázi (předpokládáme algoritmus SHA-2, 512 bitů).
   c) systém porovná řetězec vzniklý v bodě b) s otiskem hesla (viz bod 3.):
      - pokud se oba řetězce shodují, je ověření úspěšné a pokračuje se bodem 5.,
      - pokud se řetězce neshodují, znamená to, že bylo zadáno jiné heslo než to, které je uloženo v databázi a autentizační proces nebyl úspěšný. V takovém případě vrací systém hlášku „Neplatné uživatelské jméno nebo heslo“. Scénář končí.
5. **Systém zjistí, zda proběhlo přihlášení pomocí časově platného nebo exspirovaného hesla.** Za exspirované heslo je považováno heslo uživatele, kde datum a čas okamžiku přihlášení je větší než „platnost_do“ té instance objektu Heslo uživatele, pro kterou platí, že její atribut „je platné = ano“. V případě, že systém zjistí, že heslo je časově platné, je autentifikační proces úspěšně dokončen a identita uživatele ověřena.
6. V případě, že heslo je exspirované, vynutí systém na uživateli změnu hesla. Dokud nebude změna hesla úspěšně dokončena, nebude ověřena identita uživatele a proces ověření identity nebude dokončen.

4.3 Popis procesu rozšířeného potvrzení identity uživatele pomocí PINu

V některých případech bude nutné potvrdit identitu uživatele ještě dalším způsobem, než jen zadáním uživatelského jména a hesla. Typickým případem je změna hesla autoregistrovaného uživatele. V takovém případě bude aplikován následující scénář:

1. Uživatel provede akci, která vyžaduje rozšířené potvrzení identity.
2. Systém požadavek zařadí do fronty požadavků a vygeneruje pro takový požadavek jedinečný kód (PIN) dle nastavení konfiguračních položek bezpečnostní politiky. Jedinečnost takového kódu je dána pouze v rámci aktuálně existujících požadavků. Jakmile je požadavek z fronty ke zpracování vyřazen (je dokončen nebo stornován), je možno kód (PIN) znovu přidělit jinému požadavku.
3. Systém přidělí požadavku ve frontě vygenerovaný PIN a omezí platnost požadavku na určitou dobu (viz kapitola 4 Bezpečnostní politika).
4. Systém odešle PIN pomocí SMS na mobilní telefon uživatele, který akci provedl. Pokud u daného uživatele nebude mobilní telefon evidován, akci nebude možno dokončit a požadavek po čase exspiruje.
5. Uživatel přijme SMS zprávu na svém mobilním telefonu a zjistí PIN, který mu systém vygeneroval.
6. Systém nabídne uživateli zadávací pole, do kterého bude moci uživatel zapsat PIN.
7. Uživatel zadá PIN, který obdržel na svůj mobilní telefon.
8. Systém vyhodnotí, zda byl zadán PIN, který odpovídá PINu u požadavku ve frontě požadavků. Nastat mohou tyto varianty:
   a) požadavek ve frontě existuje a zadaný PIN odpovídá. Požadavek je realizován a z fronty vyřazen. Scénář končí,

b) požadavek ve frontě existuje a zadaný PIN neodpovídá. V takovém případě systém zobrazí uživateli hlášení „Nesprávný PIN“ a umožní uživateli zadat PIN znovu, případně mu umožní vygenerovat nový PIN. Požadavek ve frontě zůstává. Opakuje se bod 8.,
c) požadavek ve frontě existuje, ale uplynula doba jeho životnosti. V takovém případě systém zobrazí uživateli hlášení „Uplynula doba živostnosti požadavku, prováděná akce byla stornována“. Následně systém požadavek vyřadí z fronty. Scénář končí.

4.4 Definice bezpečnostní politiky

Aby nebylo možno prolomit hesla uživatelů hrubou silou, je třeba zajistit základní požadavky na sílu hesla (tzv. bezpečnostní politika). Protože předpokládáme, že tyto požadavky se mohou v čase měnit, je celá bezpečnostní politika navržena formou konfiguračních položek, které budou mít svoji datumovou platnost. Budou aplikovány níže uvedené konfigurační položky. V tabulce níže je vždy návrh názvu konfigurační položky, její typ, účel a defaultní hodnota

Název položky	Typ	Default	Účel
PWD_MIN_AGE	NUMBER	1440	**Minimální doba platnosti hesla (v minutách).** Účelem je zamezení rychlého vyčerpání definovaného počtu hesel tak, aby uživatel obešel kontrolu na nemožnost opakovaně použít stejné heslo. Nastavením na hodnotu 0 dojde k tomu, že tato konfigurační položka nebude zohledněna. **Default: 1 den (1440 minut)**
PWD_MAX_AGE	NUMBER	129600	**Maximální doba platnosti hesla (v minutách).** Účelem je donutit uživatele, aby po jistém čase změnil heslo, aby bylo možno vyvarovat se jeho prolomení hrubou silou (anebo se znalostí úniku otisku (hashe) hesla). **Default: 90 dní (129600 minut)**
PWD_MAX_FAILURE	NUMBER	5	**Maximální počet**

Název položky	Typ	Default	Účel
			chybných pokusů o přihlášení, po kterých je aplikační účet uživatele na určitou dobu zakázán (viz PWD_LOCK_TIME). Účelem je zamezit prolomení hesla hrubou silou strojem. **Default: 5 pokusů**
PWD_LOCK_TIME	NUMBER	20	**Počet minut, po kterých bude aplikační účet uživatele uzamčen v případě vypršení počtu povolených pokusů o přihlášení (viz PWD_MAX_FAILURE).** Účelem je zamezit prolomení hesla hrubou silou strojem. **Default: 20 minut**
PWD_FAIL_COUNT_INTERVAL	NUMBER	5	**Počet minut, po které je udržováno počítadlo chybných pokusů o přihlášení.** Účelem je umožnit uživateli získat novou sadu pokusů o přihlášení (viz PWD_MAX_FAILURE). **Default: 5 minut**
PWD_EXPIRE_WARNING	NUMBER	2520	**Určení okamžiku, kdy uživatel bude vyzván k změně hesla (počet minut před okamžikem exspirace hesla**). Účelem je informovat uživatele (zprávou na obrazovce, notifikačním e-mailem) v dostatečném předstihu před vypršením hesla, aby si jej mohl změnit a heslo mu neexspirovalo. **Default: 7 dní**

Název položky	Typ	Default	Účel
PWD_MIN_LENGTH	NUMBER	8	**Určuje minimální délku hesla (počet znaků).** Účelem je zabránění vytvoření příliš krátkých hesel, u kterých je pravděpodobnost prolomení hrubou silou možná během reálného času. **Default: 8 znaků**
PWD_MAX_LENGTH	NUMBER	20	**Určuje maximální délku hesla.** Účelem je zabránění příliš složitých hesel, kdy spíše než heslo jsou zapisovány věty, které lze snadněji uhádnout než heslo samotné. **Default: 20 znaků**
PWD_MIN_NUMERICS	NUMBER	1	**Určuje, kolik minimálně musí obsahovat heslo číslic.** Účelem je zvýšit složitost hesla, aby snadněji odolalo útoku hrubou silou. **Default: 1 číslice**
PWD_MIN_SPECIAL_CHARS	NUMBER	1	**Určuje, kolik speciálních (ne-alfanumerických) znaků musí heslo obsahovat.** Účelem je zvýšit složitost hesla, aby snadněji odolalo útoku hrubou silou. **Default: 1 speciální znak**
PWD_MIN_UPPER_CASE	NUMBER	1	**Určuje, kolik velkých písmen musí heslo obsahovat.** Účelem je zvýšit složitost hesla, aby snadněji odolalo útoku hrubou silou. **Default: 1 (heslo musí obsahovat alespoň jedno velké písmeno).**

Název položky	Typ	Default	Účel
PWD_MAX_REPEAT_CHARS	NUMBER	4	**Určuje, kolikrát se jeden znak smí maximálně vyskytovat v hesle.** Účelem je zabránění vytváření jednoduchých hesel typu ‚aaaaaaaaaa', která jsou snadno prolomitelná. **Default: 4 (jeden znak se smí opakovat maximálně 4x).**
PWD_HISTORY_DIFF_CHARS	NUMBER	2	**Určuje, v minimálně kolika znacích se musí nové heslo lišit od předcházejícího.** Účelem je zamezení vytváření jednoduchých hesel „janNovak1" „janNovak2", **Default: 2 (nové heslo se musí lišit v minimálně dvou** znacích od předcházejícího).
PWD_HISTORY_COUNT	NUMBER	10	**Určuje, kolik hesel musí být použito, aby bylo možno použít jako nové heslo již jednou použité heslo.** Účelem je zamezit neustálému používání jednoho (dvou/tří atd.) stejných hesel, což zvyšuje pravděpodobnost prolomení. **Default: 10** (stejné heslo smí být použito až po vyčerpání 10 jiných hesel).
PWD_GRACE_LIFE_TIME	**NUMBER**	**129600**	**Maximální doba životnosti hesla (v minutách) po jeho exspiraci.** Udává, jak dlouho po exspiraci hesla bude možno změnit heslo uživatelem bez jakéhokoli omezení. Po vypršení této

Název položky	Typ	Default	Účel
			doby bude možno heslo změnit pouze poté, co systém zašle do e-mailové schránky spárované s patřičným uživatelským účtem speciální URL (link), který změnu hesla umožní. Účelem je zamezení zneužití účtů, kterým exspirovalo heslo a které nejsou nadále používány. **Default: 90 dní (129600 minut).**
SMS_PIN_LENGTH	NUMBER	6	**Délka čísla automaticky generovaného systémem za účelem rozšířeného potvrzení identity (délka PINu v počtu znaků).** Účelem je dát k dispozici možnost ovlivnit délku PINu v případě, že by pevná délka nevyhovovala. **Default: 6 (6ti místné číslo).**
SMS_LIFE_TIME	NUMBER	15	**Doba životnosti PINu (v minutách).** Účelem je omezit životnost požadavků, které vyžadují rozšířené potvrzení identity uživatele na omezenou dobu. **Default: 15 minut.**

Tabulka 3 - Konfigurační položky bezpečnostní politiky

Z běžných konfiguračních položek definující bezpečnostní politiku nepožadujeme zavést:

- **definici doby, po jejímž uplynutí už nelze změnit heslo uživatelem** - vycházíme vstříc uživatelům, kteří mají nepravidelné vykazující povinnosti a mohou tak potřebovat se přihlásit k SDAT i po několika letech od posledního přihlášení,
- **automatickou exspirací účtu po nečinnosti** – vycházíme vstříc uživatelům, kteří mají nepravidelné vykazující povinnosti a mohou tak potřebovat se přihlásit k SDAT i po několika letech od posledního přihlášení.

5 Katalog funkčních požadavků

5.1.1 Uživatel, Heslo uživatele, Certifikáty uživatele

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
UMU_1.0	Vytvoření uživatele (aplikačního účtu uživatele) – interní a externí registrovaný uživatel.	Systém umožňuje vytvoření aplikačního uživatelského účtu na základě procesu popsaného v kapitole 3.1 Popis procesu vytvoření uživatelského účtu uživatelem v ČNB a souvisejících podkapitol.	Závazný	1
UMU_2.0	Změna hesla – vytvoření nového hesla a ukončení platnosti stávajícího platného hesla	Systém umožňuje měnit uživateli heslo, které se váže k jeho aplikačnímu účtu uživatele kdykoli po dobu platnosti tohoto hesla, ovšem v souladu s aktuálně platným nastavením konfiguračních položek týkajících se bezpečnostní politiky.	Závazný	1
UMU_2.1	Změna hesla – vynucení změny hesla	Systém vynucuje změnu hesla v případě, že je proveden pokus přihlásit se pomocí uživatelského aplikačního účtu, ke kterému již platnost hesla vypršela v souladu s pravidly popsanými v kapitole 3.2 Popis procesu změny hesla uživatele.	Závazný	1
UMU_3.0	Ukončení platnosti uživatele (aplikačního účtu	Systém umožňuje administrátorovi ukončit platnost existujícího aplikačního účtu uživatele. Akce ukončení platnosti aplikačního účtu uživatele bude provedena takto:	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
	uživatele)	• uživatel provede akci ukončení platnosti aplikačního uživatelského účtu, • systém před provedením akce upozorní uživatele, že tato akce je nevratná a dojde i k ukončení zařazení uživatele na všechna uživatelská místa a nabídne volby: o zrušit akci, o blokovat účet, o ukončit platnost účtu. Pokud uživatel zvolí možnost „zrušit akci“, pak systém celou akci zruší a nedochází ke změně žádných hodnot atributů. V případě, že uživatel zvolí akci „blokovat účet“, pak systém dále postupuje podle UMU_6.0 (akce ukončení platnosti aplikačního účtu uživatele není realizována). V případě, že uživatel zvolí akci „ukončit platnost účtu“, pak systém: • nastaví datum platnost_do v objektu pro sledování historie na aktuální systémový datum a čas (u instance, kde je platnost_do nastavena na maximální datum), • ke stejnému datu a času ukončí veškeré aktuálně platné zařazení uživatele na uživatelská místa (objekt Uživatel na uživatelském místě).		
UMU_4.0	Změna atributů uživatele	Systém umožňuje změnit následující atributy uživatele. • Platnost_do, • e_mail_interni_uzivatel, • dočasné_heslo (změna je možná pouze v důsledku akce „žádost	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
		o reset hesla, nikoli přímou modifikací administrátorem“, • je_aktivní (změna je možná pouze v důsledku provedení akce „aktivace účtu“, nikoli přímou modifikací administrátorem, • je_blokovaný, • jméno_uživatele, • příjmení_uživatele, • telefonní_číslo • zamčený_do_kdy.		
UMU_4.1	Změna typu uživatele	Systém neumožňuje provést změnu typu uživatele.	Závazný	2
UMU_4.2	Správa certifikátů uživatele	Systém umožňuje bez jakýchkoli omezení přidávat a odebírat certifikáty uživatele (je myšleno instance objektů Certifikát uživatele). Pro přidání certifikátu platí pravidla uvedená v kapitole 3.1.4.2 Subproces Vložení informací o elektronickém podpisu.	Závazný	2
UMU_5.0	Smazání uživatele	Systém umožňuje smazání uživatele pouze v případě, že daný uživatel není zařazen na žádném uživatelském místě. V případě, že je potřeba zakázat přístup uživateli, který je již na nějakém uživatelském místě přiřazen, bude použita funkcionalita „Blokování uživatele“ (viz UMU_6.0).	Závazný	1
UMU_6.0	Blokování uživatele	Systém umožňuje zablokovat přístup jakémukoli uživateli. V případě potřeby blokovat přístup konkrétního uživatele, administrátor nastaví atribut „je_blokovaný“ dané instance objektu Uživatel na hodnotu „ano“. V případě, že je daný uživatel právě přihlášený k systému, systém	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
		jej automaticky odhlásí a neumožňuje mu znovu se přihlásit. Na e-mailovou adresu uživatele, kterému byl zablokován aplikační účet uživatele, je odeslána e-mailová zpráva obsahující informaci o tom, že přístup do aplikace SDAT byl zablokován správcem systému a informace o kontaktní osobě, na kterou je možno se obrátit s reklamací daného postupu.		
UMU_6.1	Odblokování uživatele	Systém umožňuje odblokovat přístup pro jakéhokoli zablokovaného uživatele. V takovém případě administrátor nastaví atribut „je_blokovaný“ dané instance objektu Uživatel na hodnotu „ne“. Na e-mailovou adresu uživatele, kterému byl odblokován aplikační účet uživatele, je odeslána e-mailová zpráva obsahující informaci o tom, že přístup do aplikace SDAT byl odblokován.	Závazný	1
UMU_7.0	Obnovení (reset) zapomenutého hesla – externí registrovaný uživatel	Systém umožňuje externímu registrovanému uživateli provést reset (obnovení) existujícího zapomenutého hesla na základě procesu popsaného v kapitole 3.3 Popis procesu resetování hesla uživatele a souvisejících podkapitol.	Závazný	2
UMU_9.0	Ověření identity uživatele – externí registrovaný uživatel	Systém ověřuje identitu externího registrovaného uživatele v souladu s procesem popsaným v kapitole 3.4 Popis procesu autentizace — externí registrovaný uživatel.	Závazný	2
UMU_10.0	Ověření identity uživatele – interní uživatel (SSO	Systém ověřuje identitu interního uživatele v souladu s procesem popsaným v kapitole 3.5 Popis procesu autentizace — interní uživatel.	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
	autentifikace)			
UMU_11.0	Zamčení aplikačního účtu uživatele – externí registrovaný uživatel	V případě, že uživatel zadá chybně heslo, systém počítá počet těchto chybných pokusů. V případě, že počet chybných pokusů jednoho uživatele přesáhne povolený počet chybných pokusů (povolený počet chybných pokusů je uveden v konfigurační položce **PWD_MAX_FAILURE**), pak systém daný aplikační účet uživatele uzamkne (na dobu uvedenou v konfigurační položce **PWD_LOCK_TIME**). Zamčení provede tak, že uvede do atributu „zamčený_do_kdy“ aktuální čas navýšený o počet minut uvedený v konfigurační položce **PWD_LOCK_TIME.** Systém vynuluje počítadlo chybných pokusů po určité době od prvního chybného přihlášení. Tato doba je stanovena konfigurační položkou **PWD_FAIL_COUNT_INTERVAL**. Tento požadavek platí pouze pro případ, že je zároveň poskytnuto existují ID externího uživatele.	Závazný	2
UMU_11.1	Odemknutí uzamčeného aplikačního účtu	Systém umožňuje uživateli (administrátorovi) odemknout uzamčený aplikační účet, který byl uzamčen v důsledku překročení maximálního povoleného počtu pokusů o přihlášení.	Závazný	1
UMU_12.0	Notifikace uživatele o blížícím se vypršení hesla – e-mail	Systém jedenkrát denně (v době mimo špičku) spouští proceduru, která provádí kontrolu platnosti hesel u všech registrovaných externích uživatelů. Kontrolu provádí pouze u hesel aplikačních účtů, které splňují následující podmínky: • aplikační účet uživatele je aktivní (atribut „je_aktivní = ano“),	Závazný	2

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
		• aplikační účet uživatele je typu „externí registrovaný“, tedy atribut „typ uživatele = externí“ a atribut „je_registrovaný = ano“, • aplikační účet uživatele není blokovaný, tedy atribut „je_blokovaný = ne“. V případě, že zjistí, že aktuálně platné heslo aplikačního účtu uživatele, které splňuje podmínky uvedené výše, vyprší v době, která je menší než počet dní uvedený v konfigurační položce **PWD_EXPIRE_WARNING**, pak systém odešle e-mailovou zprávu na e-mailovou adresu uživatele, která bude obsahovat informaci o tom, že se blíží termín vypršení hesla (a uvede, kdy přesně heslo vyprší).		
UMU_12.1	Notifikace uživatele o blížícím se vypršení hesla – zobrazení v aplikaci	Systém uživateli, kterému se blíží vypršení hesla podle definice v UMU_12.0, zobrazí tuto informaci jako varování ve stavovém řádku aplikace (nebo jiným podobným způsobem) a zároveň umožňuje přímo z tohoto varování přejít na formulář pro změnu hesla.	Závazný	2
UMU_13.0	Autoregistrace	Systém umožňuje vytvořit v systému novou Osobu a v rámci ní aplikační účet uživatele a hlavní uživatelské místo bez zásahu jakéhokoli uživatele na straně ČNB. Celý proces systém vykoná v souladu s podmínkami uvedenými v kapitole 3.7 Popis procesu vytvoření externí Osoby a aplikačního účtu uživatele externím subjektem (Autoregistrace).	Závazný	2
UMU_14.0	Dvoukroková autentifikace	Systém umožňuje provádění dvoukrokové autentifikace (dodatečného ověření identity). Tato autentifikace probíhá u určených akcí podle scénáře uvedeného v kapitole 4.3 Popis procesu rozšířeného potvrzení identity uživatele pomocí PINu.	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
UMU_15.0	Postup získání a vyhodnocení oprávnění	Systém přiděluje uživateli taková oprávnění, jaká získá zařazením uživatele na uživatelská místa v souladu s algoritmem uvedeným v kapitole 2.9.13 Postup získání a vyhodnocení oprávnění.	Závazný	1
UMU_16.0	Zobrazení uživatelů, uživatelských míst a oprávnění	Systém zobrazuje na jedné obrazovce následující údaje: • seznam všech uživatelů. Tento seznam je možno filtrovat podle: o typu uživatele – systém nabízí tyto volby: ▪ všichni uživatelé, ▪ interní uživatelé, ▪ externí registrovaní uživatelé, ▪ externí neregistrovaní uživatelé, o aktivity uživatele – systém nabízí tyto volby: ▪ všichni uživatelé, ▪ pouze aktivní uživatelé, ▪ pouze neaktivní uživatelé, o zamčení uživatelé – systém nabízí tyto volby: ▪ všichni uživatelé, ▪ pouze zamčení uživatelé, o blokovaní uživatelé ▪ všichni uživatelé, ▪ pouze neblokovaní uživatelé, ▪ pouze blokovaní uživatelé. Pokud uživatel definuje hodnoty ve více filtrovacích kritériích, systém spojí tato filtrovací kritéria logickým operátorem „AND“. Nastavení filtrovacích kritérií systém zjišťuje z uživatelské konfigurace; pokud není uživatelská konfigurace k dispozici, pak systém zobrazí defaultní	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
		nastavení filtrů, což znamená, že u všech filtrovacích kritérií je vybrána hodnota „všichni uživatelé“. Systém umožňuje uživateli vybrat právě jednoho uživatele a provést s ním povolené akce definované výše uvedenými funkčními požadavky. K vybranému uživateli systém zobrazí seznam všech uživatelských míst, na kterých je přiřazen. Systém zobrazuje: • datum a čas začátku platnosti přiřazení uživatele na uživatelské místo, • datum a čas konce platnosti přiřazení uživatele na uživatelské místo, • název a typ uživatelského místa, • název Osoby, ke které je uživatelské místo připojeno (pokud vazba na Osobu u uživatelského místa existuje). Seznam uživatelských míst je možno filtrovat za použití kritérií „všechna uživatelská místa“ NEBO „pouze aktuální uživatelská místa“. Nastavení filtrovacích kritérií systém zjišťuje z uživatelské konfigurace; pokud není uživatelská konfigurace k dispozici, pak systém zobrazí defaultní nastavení filtrů, což znamená, že u všech filtrovacích kritérií je vybrána hodnota „všechna uživatelská místa“. Systém umožňuje uživateli vybrat právě jedno uživatelské místo a provést s ním povolené akce definované níže uvedenými funkčními požadavky. Zároveň umožňuje uživateli přejít do zobrazení, kde je primárním objektem uživatelské místo (viz UMM_11.0) a zde zobrazit detail tohoto vybraného uživatelského místa. Systém umožňuje zobrazit oprávnění (získání definice oprávnění je		

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
		popsáno funkčním požadavkem UMU_15.0) v následujícím rozsahu: ○ pro vybraného uživatele, za všechna jeho uživatelská místa s možností definice, zda se má jednat o všechna uživatelská místa nebo pouze místa platná k určitému datu/času (defaultně se jako datum/čas v tomto případě použije aktuální systémové datum), ○ pro vybraného uživatele a vybrané uživatelské místo. Systém umožňuje zobrazit oprávnění: • seskupené za osobu/výkaz, tj. soupis aktivit, které jsou v rámci daného oprávnění povoleny pro danou osobu/výkaz s vyznačením datových oblastí, pro které toto povolení platí pro případ, že jsou definovány negativní výjimky, • seskupené za výkaz/osobu, tj. soupis aktivit, které jsou v rámci daného oprávnění povoleny pro danou osobu/výkaz s vyznačením datových oblastí, pro které toto povolení platí pro případ, že jsou definovány negativní výjimky, • seskupené za aktivitu, tj. soupis Osob a Výkazů, které jsou pro danou aktivitu povoleny.		
UMU_18.0	Přístup neregistrovaného uživatele	Systém umožňuje přístup (přihlášení) neregistrovaného uživatele podle pravidel popsaných v kapitole 3.6 Popis procesu přístupu neregistrovaného uživatele.	Závazný	1
UMU_19.0	Přístup externích registrovaných uživatelů k odeslaným datům	Systém provede dvoukrokovou autentifikaci (viz kapitola 4.3 Popis procesu rozšířeného potvrzení identity uživatele pomocí PINu) v případě, že se k aplikaci SDAT přihlásí externí registrovaný uživatel a pokusí ze zobrazit historicky vykázaná data.	Závazný	2

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
		Proces rozšířeného ověření identity bude aplikován v následujících případech: • uživatel se pokusí zobrazit vykázaná data (hodnoty údajů) za Výkaz, který je označen jako citlivý, • uživatel se pokusí zobrazit obsah Vstupní zprávy, která obsahuje alespoň jeden citlivý Výkaz Pokud proces rozšířeného ověření identity bude úspěšně dokončen, tj. uživatel prokáže svoji identitu rozšířeným způsobem prokazování identity, pak systém uživateli poskytne daná data. Proces rozšířeného ověření identity je možno provést pouze jednou v rámci tzv. session. Se zánikem session dojde k zániku tzv. credentials, a je nutno se znovu autentifikovat. V takovém případě a v případě nové žádosti o citlivá data je nutno opakovat celý proces rozšířeného ověření identity.		

5.1.2 Uživatelská místa, Rozsah oprávnění a Uživatel na uživatelském místě

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
UMM_1.0	Vytvoření uživatelského místa	Systém umožňuje vytvořit uživatelské místo. Musí být splněny tyto podmínky: • vytvoření uživatelského místa je dovoleno pouze interním a externím registrovaným uživatelům. Externím neregistrovaným	Závazný	1

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie

		uživatelům není vytváření uživatelských míst žádným způsobem umožněno (těmto uživatelům jsou uživatelská místa vytvářena automaticky systémem),
		• pokud je uživatelské místo vytvářeno externím registrovaným uživatelem, pak je vždy vytvářeno v rámci právě jedné Osoby (takové Osoby, pro kterou je vytvořeno hlavní UM). Atribut „typ_uživatelského_místa“ je nastaven na hodnotu „externí“,
		• pokud je uživatelské místo vytvářeno externím registrovaným uživatelem, nemůže být označeno jako hlavní (hlavní UM může být vytvořeno pouze procesem autoregistrace a nebo založením Osoby interním uživatelem),
		• uživatelské místo zakládané interním uživatelem může být založeno jako interní i externí:
		  ○ v případě, že je zakládáno jako externí (atribut „typ uživatelského místa = externí“), pak toto uživatelské místo musí být spojeno s právě jednou Osobou. V případě zakládání externího uživatelského místa interním uživatelem je možno založit uživatelské místo jako hlavní (atribut „je hlavní = ano“), pouze za předpokladu, že pro danou Osobu neexistuje jiné hlavní uživatelské místo,
		  ○ v případě, že je uživatelské místo zakládáno jako interní (atribut „typ uživatelského místa = interní“), pak toto uživatelské místo nesmí být napojeno na žádnou Osobu. Atribut „je hlavní“ je nastaven vždy na hodnotu „ne“,
		• uživatelské místo je vždy založeno jako aktivní (atribut „je aktivní = ano“),
		• systém automaticky nastaví časovou platnost uživatelského místa		

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
		tak, že: ○ atribut „platnost_od“ nastaví automaticky (bez možnosti změnit tento údaj uživatelem) na systémové datum a čas (na úroveň sekund), kdy je uživatelské místo založeno ○ atribut „platnost_do“ nastaví automaticky na maximální datum a čas (na úroveň sekund) s možností změny uživatelem za splnění podmínky, že atribut „platnost_do“ nebude menší než atribut „platnost_od“. • název uživatelského místa musí být jedinečný v rámci: ○ typu uživatelského místa u interních uživatelských míst, ○ typu uživatelského místa a Osoby u externích uživatelských míst.		
UMM_2.0	Změna atributů uživatelského místa	Systém umožňuje změnit následující atributy vybrané instance objektu Uživatelské místo: • atribut „název_uživatelského_místa“ (nový název nesmí být duplicitní; viz podmínky pro vytvoření uživatelského místa a UMM_1.0); tato změna je dovolena jak interním tak i externím uživatelům. Interní uživatel může měnit název jakéhokoli uživatelského místa, externí jen u uživatelských míst v rámci své Osoby (mimo hlavního uživatelského místa), • atribut „je_aktivní“; tato změna je dovolena jak interním, tak i externím uživatelům. Interní uživatel může aktivitu jakéhokoli uživatelského místa, externí jen aktivitu u uživatelských míst v rámci své Osoby (mimo hlavního uživatelského místa). Nastavení atributu na „ne“ znamená, že dané uživatelské místo není možno použít pro přístup do systému. Slouží pro **dočasné** vyřazení uživatelského místa	Závazný	1

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
		z používání, • atribut „platnost_do“; tato změna je dovolena jak interním, tak i externím uživatelům. Interní uživatel může měnit platnost_do jakéhokoli uživatelského místa, externí jen u uživatelských míst v rámci své Osoby (mimo hlavního uživatelského místa). Atribut „platnost_do“ obsahuje časovou složku definovanou na úroveň sekund a nesmí být nižší než atribut „platnost_od“. Atribut „platnost_do“ slouží pro **trvalé** vyřazení uživatelského místa z používání. • atribut poznámka, • atribut popis. Systém nedovoluje změnit hodnotu atributu „platnost_do“ v případě, že je tato hodnota menší než aktuální systémové datum a čas (na úroveň sekund). V případě, že uživatel mění hodnotu atributu „platnost_do“, pak systém neumožňuje, aby toto datum/čas bylo menší než je aktuální systémové datum a čas, kdy ke změně atributu došlo (na úroveň sekund). Hodnoty atributů „typ_uživatelského_místa“, „je hlavní“ a „platnost_od“ nemohou být změněny.		
UMM_3.0	Smazání uživatelského místa	Systém umožňuje uživateli smazat uživatelské místo pouze v případě, že k tomuto uživatelskému místu nebyli doposud přiřazeni uživatelé. V případě, že dané uživatelské místo je třeba vyřadit z používání, je třeba provést ukončení jeho platnosti pomocí změny platnosti uživatelského místa (viz UMM_2.0). Důvodem pro nemožnost smazání uživatelského místa, které již bylo použito/mohlo být použito pro přístup uživatele, je přístup k plné historii uživatelských míst s ohledem na	Závazný	1

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
		nutnost mít komplexní přehled o tom, jak byla historicky nastavena oprávnění pro potřeby auditu.		
UMM_4.0	Zařazení uživatele na uživatelské místo	Systém umožňuje administrátorovi zařadit uživatele na uživatelské místo. Zařazením uživatele na uživatelské místo získává uživatel oprávnění, která jsou k danému uživatelskému místu přiřazena. Pro přiřazení uživatele na uživatelské místo platí tato pravidla: • typ uživatele a typ uživatelského místa musí být v souladu. To znamená, že interního uživatele lze přiřadit pouze na interní uživatelské místo, stejně tak jako externího registrovaného uživatele lze přiřadit jen na externí uživatelské místo, • přiřazení uživatele na uživatelské místo se děje vždy s vymezením časové platnosti tohoto zařazení. Systém automaticky nastaví časový úsek platnosti zařazení uživatele na uživatelské místo tak, že: o platnost_od nastaví automaticky na systémové datum a čas (na úroveň sekund), kdy je uživatel na uživatelské místo zařazen, Systém dovolí uživateli toto datum změnit na jakékoli budoucí datum pro možnost dopředu definovat dočasný zástup. o platnost_do nastaví automaticky na maximální datum a čas (na úroveň sekund) s možností změny uživatelem, za splnění podmínky, že platnost_do nebude menší než platnost_od. Datum a čas platnosti zařazení uživatele na uživatelské místo nesmí vystoupit z rámce platnosti uživatelského místa, na které je uživatel zařazován. Jeden uživatel smí být na jedno uživatelské místo zařazen v jeden	Závazný	1

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
		okamžik maximálně jednou.		
UMM_5.0	Ukončení zařazení uživatele na uživatelské místo	Systém umožňuje ukončit zařazení uživatele na uživatelské místo (objekt Uživatel na uživatelském místě). Musejí být splněna následující pravidla: • externí registrovaný uživatel může ukončovat zařazení uživatele na uživatelské místo pouze u těch uživatelských míst, která jsou označena jako externí a jsou založena v rámci jeho Osoby a která nejsou označena jako hlavní uživatelské místo, • interní uživatel může ukončovat zařazení uživatele na uživatelské místo pouze u těch uživatelských míst, která jsou označena jako interní. U externích uživatelských míst smí ukončovat zařazení uživatelů pouze na UM, které je označeno jako hlavní. Platnost_do platnosti zařazení uživatele na uživatelské místo nesmí být menší než hodnota atributu „platnost_od“ daného zařazení a zároveň nesmí být menší než aktuální systémové datum a čas (na úroveň sekund) v okamžiku ukončení. Systém nedovoluje změnit hodnotu atributu „platnost_do“ v případě, že je tato hodnota menší než aktuální systémové datum a čas (na úroveň sekund), tj. jednou ukončené zařazení uživatele na uživatelské místo nelze prodloužit.	Závazný	1
UMM_5.1	Změna zařazení uživatele na uživatelské místo	Systém umožňuje editaci zařazení uživatele na uživatelské místo za splnění těchto pravidel: • Změna hodnoty atributu „platnost_od“ je možná jen tehdy, pokud je aktuální systémové datum menší, než hodnota atributu	Závazný	1

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
		„platnost_od“. Nová hodnota nesmí být nižší než je aktuální systémové datum a zároveň nesmí být větší než hodnota atributu „platnost_do“. • Změna hodnoty atributu „platnost_do“ je možná tehdy, pokud je hodnota tohoto atributu, větší než je aktuální systémové datum. Nová hodnota nesmí být menší než je aktuální systémové datum a zároveň musí být větší než je hodnota atributu „platnost_od“.		
UMM_6.0	Smazání zařazení uživatele na uživatelské místo	Systém neumožňuje smazat žádné zařazení uživatele na uživatelské místo. V případě, že je třeba ukončit zařazení uživatele na uživatelské místo a odebrat mu tak oprávnění, je třeba ukončit platnost zařazení uživatele na uživatelském místě (viz UMM_5.0). Důvodem pro nemožnost smazání zařazení uživatele na uživatelské místo je přístup k plné historii nastavení oprávnění pro potřeby auditu.	Závazný	1
UMM_7.0	Vytvoření rozsahu oprávnění	Systém umožňuje pro každé uživatelské místo vytvořit neomezený počet rozsahu oprávnění (instancí objektu Rozsah oprávnění). Každá instance objektu Rozsah oprávnění má definovánu svoji časovou platnost (na úroveň sekund). Datum a čas platnosti instance objektu Rozsah oprávnění nesmí vystoupit z rámce platnosti uživatelského místa, ke kterému je tento rozsah oprávnění definován. Pro definici rozsahu oprávnění platí pravidla uvedená v kapitole 2.9 Objekt Rozsah oprávnění a všech jejich podkapitolách a dále systém automaticky nastaví časový úsek platnosti rozsahu oprávnění tak, že • platnost_od nastaví automaticky (bez možnosti změnit tento údaj uživatelem) na systémové datum a čas, kdy je uživatelské místo	Závazný	1

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
		založeno (na úroveň sekund), • platnost_do nastaví automaticky na maximální datum a čas (na úroveň sekund) s možností změny uživatelem, za splnění podmínky, že atribut „platnost_do“ nebude menší než atribut „platnost_od“. V případě definice rozsahu oprávnění pro externí Osobu je v nabídce Výkazů použita jen taková množina Výkazů, pro které je pro danou Osobu nadefinována vykazovací povinnost.		
UMM_8.0	Změna rozsahu oprávnění	Systém umožňuje změnit existující rozsah oprávnění. Pro změnu rozsahu oprávnění platí stejná pravidla jako pro vytváření rozsahu oprávnění (viz UMM_7.0). V případě, že uživatel změní rozsah oprávnění, je původní instance objektu Rozsah oprávnění ukončena k datu a času vzniku nového oprávnění minus 1 sekunda. Pro změnu rozsahu oprávnění platí tato pravidla: • interní uživatel může měnit rozsah oprávnění jakéhokoli uživatelského místa, • externí registrovaný uživatel může měnit rozsah oprávnění pouze u externích uživatelských míst a zároveň míst, která jsou založena v rámci jeho Osoby. Musí se však jednat o uživatelská místa, která nejsou označena jako hlavní.	Závazný	1
UMM_9.0	Smazání rozsahu oprávnění	Systém neumožňuje smazat žádný existující rozsah oprávnění. V případě, že je třeba ukončit nějaký existující rozsah oprávnění, je třeba ukončit platnost rozsahu oprávnění (viz UMM_10.0). Důvodem pro nemožnost smazání oprávnění je přístup k plné historii nastavení oprávnění pro potřeby auditu.	Závazný	1

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
UMM_10.0	Ukončení rozsahu oprávnění	Systém umožňuje uživateli ukončit jakýkoli existující rozsah oprávnění. Musí však být splněno, že hodnota atributu „platnost_do“ nebude nižší, než hodnota atributu „platnost_od“ a zároveň nebude nižší, než je aktuální systémové datum v době provádění ukončení. Pro změnu rozsahu oprávnění platí tato pravidla: • interní uživatel může měnit rozsah oprávnění jakéhokoli uživatelského místa. • externí registrovaný uživatel může měnit rozsah oprávnění pouze u externích uživatelských míst a zároveň míst, která jsou založena v rámci jeho Osoby. Musí se však jednat o uživatelská místa, která nejsou označena jako hlavní.	Závazný	1
UMM_11.0	Zobrazení uživatelských míst, oprávnění a uživatelů	Systém zobrazuje na jedné obrazovce následující údaje: • seznam všech uživatelských míst s možností filtrovat tento seznam na uživatelská místa platná k určitému datu, • pro vybrané uživatelské místo seznam všech uživatelů, kteří jsou na daném uživatelském místě přiřazeni, s možností filtrovat tento seznam na uživatele, kteří jsou přiřazeni na uživatelské místo s platností k určitému datu. Systém umožňuje uživateli přejít do zobrazení, kde je primárním objektem uživatel (viz UMU_16.0) a zde zobrazit detail tohoto vybraného uživatele.	Závazný	1
UMM_12.0	Detailní zobrazení rozsahu oprávnění	Systém zobrazuje rozpis oprávnění: • v takovém rozpisu, v jakém jsou data zadaná, tedy jako instance objektů Osoba/Výkaz a Typ Osoby/Vykazovací rámec a Definice výjimky z rozsahu oprávnění,	Závazný	1

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
		• v detailním rozpisu, kdy je zobrazen už výsledek vyhodnocení oprávnění ve formě kombinací Výkaz/Osoba. V tomto případě nejsou zobrazovány instance objektů, které umožňují dynamickou definici rozsahu oprávnění a nejsou zobrazovány ani výjimky z této dynamické definice; zobrazována je pouze výsledná množina kombinací Osoba/Výkaz, ke kterým má uživatel (uživatelské místo) povolen přístup.		
UMM_13.0	Vygenerování žádosti o přidělení oprávnění	Systém umožňuje vygenerovat žádost o přidělení oprávnění v souladu s procesem a omezujícími podmínkami popsaným v kapitole 3.8 Proces Přidělení oprávnění pro přístup k datům.	Závazný	3
UMM_14.0	Filtrování závislých výběrových seznamů – definování žádosti	Pokud žadatel při tvorbě žádosti o přístup k datům definuje Výkaz nebo Vykazovací rámec, systém odpovídajícím způsobem zužuje nabídku na Vykazující osobu (Typ osoby, Vykazující osoba) a naopak.	Závazný	3
UMM_15.0	Evidence útvarů – přístup k datům bez souhlasu vlastníka Výkazu	Systém umožňuje uživateli evidovat útvary, pro které není potřeba v žádosti o přístup k necitlivým datům dávat souhlas vlastníka dat.	Závazný	3
UMM_16.0	Zobrazení a tisk přístupových práv	Systém umožňuje uživateli zobrazit a vytisknout za sebe přístupová práva k datům a metadatům.	Závazný	3
UMM_17.0	Zobrazení a tisk přístupových práv podřízených zaměstnanců	Systém umožňuje uživateli (vedoucímu zaměstnanci) zobrazit a vytisknout přístupová práva k datům a metadatům za uživatele, kteří jsou mu podřízeni dle Řídicí databáze ČNB.	Závazný	3

ID požadavku	Název požadavku	Popis požadavku	Důležitost	Kategorie
UMM_18.0	Filtrování závislých výběrových seznamů – definování přístupu k datům	Pokud administrátor při zadávání přístupu k datům definuje Výkaz nebo Vykazovací rámec, systém odpovídajícím způsobem zužuje nabídku na Vykazující osobu (Typ osoby, Vykazující osoba) a naopak.	Závazný	3
UMM_19.0	Výběr omezené množiny UM pro práci	Standardě platí, že uživateli jsou nastavena oprávnění podle všech platných a aktivních uživatelských míst, na kterých je přiřazen. Systém umožňuje uživateli vybrat z jemu přiřazených UM pouze omezenou množinu UM, pod kterými bude následně pracovat. Systém toto umožňuje uživateli takovým způsobem, aby nebylo nutné se odhlašovat a znovu přihlašovat. Po výběru této omezené množiny systém znovu vyhodnotí oprávnění a nastaví pouze taková oprávnění, jaká jsou spjata s vybranou množinou UM. Tímto postupem má být dosaženo toho, aby bylo možno se vyhnout situaci, kdy zařazením uživatele na další UM dojde k tomu, že uživateli budou práva odebrána (nové UM obsahuje taková oprávnění (zápornou výjimku), které vyhrají nad oprávněními nad dosavadními UM.	Závazný	1

5.1.3 Role a Aktivity

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
RAK_1.0	Vytvoření nové (aplikační) aktivity	Systém umožňuje vytvoření nové (aplikační) aktivity. Každá instance objektu Aplikační aktivita může obsahovat neomezeně systémových aktivit. Každá nová instance objektu Aktivita je založena jako aktivní (hodnota atributu „je_aktivní = ano“).	Závazný	1
RAK_2.0	Změna existující aktivity	Systém umožňuje měnit uživateli hodnotu atributu „je_aktivní“. Systém neumožňuje měnit jednoznačnou identifikaci aktivity.	Závazný	1
RAK_3.0	Smazání existující aktivity	Systém neumožňuje smazání žádné existující instance objektu Aktivita; v případě potřeby vyřadit nějakou aktivitu (funkčnost není dále implementovaná, je nutno funkčnost dočasně zakázat), je nutno provést změnu existující aktivity (viz RAK_2.0).	Závazný	1
RAK_4.0	Vytvoření nové role	Systém umožňuje vytvoření nové role. Každá nová instance objektu Role je založena jako aktivní (hodnota atributu „je_aktivní = ano“). Název role musí být jedinečný (v rámci daného jazyka). Role může být založena samostatně (bez vazby na žádnou jinou roli) anebo s vazbou na právě jednu další roli (viz rekurzivní vazba Rodičovská role v objektovém modelu). V takovém případě se jedná o vytvoření hierarchie rolí, kdy nově založená role je potomkem existující rodičovské role.	Závazný	1
RAK_5.0	Změna existující role	Systém umožňuje změnit existující instanci objektu Role v rozsahu textových charakteristik (název, popis, poznámka) a u atributu „typ_zařazení“.	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
		Systém umožňuje měnit uživateli hodnotu atributu „je_aktivní“. V případě jakékoli změny existující instance Aktivita provede systém její zaverzování. Systém neumožňuje měnit jednoznačnou identifikaci aktivity.		
RAK_6.0	Smazání existující role	Systém umožňuje smazat roli pouze v případě, že k ní nejsou připojeny žádné aktivity a v případě, že daná role není přiřazena (přímo nebo přes nadřízenou roli) k žádnému uživatelskému místu.; v případě potřeby vyřadit nějakou roli, systém umožňuje uživateli změnit příznak „je_aktivní“ na hodnotu „ne“.	Závazný	1
RAK_7.0	Zařazení aktivity do role	Systém umožňuje zařazovat aktivity do role. Jedna aktivita smí být zařazena do více rolí a jedna role smí mít přiřazeno více aktivit. Jedna aktivita smí být přiřazena jedné roli maximálně jednou v jeden časový okamžik. Zařazení aktivity do role se děje vždy s vymezením časové platnosti tohoto zařazení. Systém automaticky nastaví časový úsek platnosti zařazení aktivity do role tak, že: • platnost_od nastaví automaticky (bez možnosti změnit tento údaj uživatelem) na systémové datum a čas (na úroveň sekund), kdy je aktivita do role zařazována) • platnost_do nastaví automaticky na maximální datum a čas (na úrovni sekund) s možností změny uživatelem, za splnění podmínky, že platnost_do nebude menší než platnost_od.	Závazný	1
RAK_8.0	Ukončení zařazení aktivity do role	Systém umožňuje ukončit zařazení aktivity do role (objekt Aktivita v roli), pokud je splněno pravidlo, že platnost_do zařazení aktivity do role	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
		nesmí být menší než hodnota atributu „platnost_od“ daného zařazení a zároveň nesmí být menší než aktuální systémové datum a čas (na úroveň sekund) v okamžiku ukončení. Systém nedovoluje změnit hodnotu atributu „platnost_do“ v případě, že je tato hodnota menší než aktuální systémové datum a čas (jednou ukončené zařazení aktivity do role nelze prodloužit).		
RAK_9.0	Změna zařazení aktivity do role	Systém nedovoluje měnit instanci objektu Aktivita v roli s výjimkou ukončení platnosti (viz RAK_8.0).	Závazný	1
RAK_10.0	Smazání zařazení aktivity do role	Systém neumožňuje smazat žádné zařazení aktivity do role. V případě, že je třeba ukončit zařazení aktivity do role, je třeba ukončit platnost instance objektu Aktivita v roli (viz RAK_8.0). Důvodem pro nemožnost smazání zařazení aktivity do role je přístup k plné historii nastavení oprávnění pro potřeby auditu.	Závazný	1
RAK_11.0	Přiřazení role k uživatelskému místu	Systém umožňuje přiřazovat role k uživatelskému místu. Jedna role smí být přiřazena k více uživatelským místům a jedno uživatelské místo smí mít přiřazeno více rolí. Jedna role smí být přiřazena jednomu uživatelskému místu maximálně jednou v jeden časový okamžik. Přiřazení role k uživatelskému místu se děje vždy s vymezením časové platnosti tohoto zařazení. Systém automaticky nastaví časový úsek platnosti zařazení aktivity do role tak, že: • platnost_od nastaví automaticky (bez možnosti změnit tento údaj uživatelem) na systémové datum a čas (na úroveň sekund), kdy je role k uživatelskému místu přiřazena, • platnost_do nastaví automaticky na maximální datum a čas (na	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
		úroveň sekund) s možností změny uživatelem, za splnění podmínky, že platnost_do nebude menší než platnost_od. Datum a čas platnosti zařazení role na uživatelské místo nesmí vybočit z platnosti uživatelského místa, kam je role přiřazována.		
RAK_12.0	Ukončení přiřazení role k uživatelskému místu	Systém umožňuje ukončit přiřazení role k uživatelskému místu (objekt Role na uživatelském místě), pokud je splněno pravidlo, že platnost_do přiřazení role k uživatelskému místu nesmí být menší než hodnota atributu „platnost_od“ daného zařazení a zároveň nesmí být menší než aktuální systémové datum a čas (na úroveň sekund) v okamžiku ukončení. Systém nedovoluje změnit hodnotu atributu „platnost_do“ v případě, že je tato hodnota menší než aktuální systémové datum a čas (na úroveň sekund), tj. jednou ukončené přiřazení role k uživatelskému místu nelze prodloužit.	Závazný	1
RAK_13.0	Změna zařazení role na uživatelské místo	Systém nedovoluje měnit instanci objektu Role na uživatelském místě s výjimkou ukončení platnosti (viz RAK_12.0).	Závazný	1
RAK_14.0	Smazání přiřazení role k uživatelskému místu	Systém neumožňuje smazat žádné přiřazení role k uživatelskému místu. V případě, že je třeba ukončit přiřazení role k uživatelskému místu, je třeba ukončit platnost přiřazení role k uživatelskému místu (viz RAK_12.0). Důvodem pro nemožnost smazání přiřazení role k uživatelskému místu je přístup k plné historii nastavení oprávnění pro potřeby auditu.	Závazný	1
RAK_15.0	Zobrazení rolí a	Systém zobrazuje na jedné obrazovce následující údaje:	Závazný	1

ID požadavku	Název požadavku	Popis Požadavku	Důležitost	Kategorie
	aktivit	• seznam všech rolí jako „graf“ (grafem se rozumí zobrazení rolí v hierarchické struktuře, nicméně nemůže se jednat o strom, protože v tomto případě může existovat více kořenů stromu), který bude respektovat nadřízenost a podřízenost jednotlivých rolí, • na základě vybrané role umožňuje zobrazit seznam všech aktivit, které jsou k dané roli přiřazeny s možností odfiltrovat seznam pouze na ty aktivity, které jsou přiřazeny ke konkrétnímu datu/času, ať už jsou tyto aktivity přiřazeny k roli přímo nebo jsou přiřazeny nějaké roli, která je potomkem právě vybrané role. Systém vizuálně odliší aktivity, které jsou přiřazeny k roli přímo a aktivity, které jsou přiřazeny na základě přiřazení k nějaké podřízené roli a umožňuje uživateli podle těchto dvou hledisek filtrovat, • na základě vybrané role zobrazí seznam všech uživatelských míst, kterým je daná role přiřazena s možností odfiltrovat tento seznam na uživatelská místa, která jsou k roli přiřazena s platností k určitému datu a času. Systém umožňuje uživateli vybrat právě jedno uživatelské místo a přejít do zobrazení, kde je primárním objektem uživatelské místo, viz UMM_11.0 a zde zobrazit detail tohoto vybraného uživatelského místa.		