FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE

ZADÁNÍ DIPLOMOVÉ PRÁCE

Název:	Implementace systému GDPR Podatelna na platformě Salesforce
Student:	Bc. Jan Bláha
Vedoucí:	Ing. Jan Svatoš, Ph.D.
Studijní program:	Informatika
Studijní obor:	Webové a softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2018/19

Pokyny pro vypracování

Seznamte se s obecným nařízením Evropského parlamentu 2016/679 *General Data Protection Regulation* (GDPR), které upravuje a zpřísňuje pravidla pro nakládání s osobními údaji v organizacích.
Proveďte analýzu požadavků na systém "GDPR Podatelna", který bude poskytovat prostředí pro obsluhu GDPR žádostí.
Proveďte rešerši existujících nástrojů podporujících správu GDPR žádostí.
Zmapujte technické aspekty platformy Salesforce a zohledněte je v analýze požadavků.
Systém naimplementujte jako modul pro platformu Salesforce, který bude možné publikovat v obchodě aplikací AppExchange.
Modul navrhněte tak, aby byl v budoucnu snadno rozšiřitelný a řádně ho zdokumentujte.
Pro vytvořený systém zvolte metodiku testování a testování proveďte.
Zhodnoťte použitelnost aplikace a navrhněte případná vylepšení.

Seznam odborné literatury

Dodá vedoucí práce.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
děkan

V Praze dne 29. ledna 2018

FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE

Diplomová práce

Implementace systému GDPR Podatelna na platformě Salesforce

Bc. Jan Bláha

Katedra softwarového inženýrství
Vedoucí práce: Ing. Jan Svatoš, Ph.D.

7. května 2018

Poděkování

Děkuji vedoucímu diplomové práce Ing. Janu Svatošovi, Ph.D. za cenné rady a odborné vedení této práce.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 7. května 2018 .....

Odkaz na tuto práci

Bláha, Jan. *Implementace systému GDPR Podatelna na platformě Salesforce*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2018.

Abstrakt

Tato diplomová práce se zabývá návrhem a implementací systému GDPR Podatelna na cloudové platformě Salesforce. Systém slouží k obsluze a zpracování žádostí, které nově mohou podávat občané Evropské unie na základě obecného nařízení Evropského parlamentu *General Data Protection Regulation* (GDPR).

Systém byl vyvinut pomocí standardních nástrojů platformy Salesforce a byl navrhnut tak, aby jej bylo možné v budoucnu publikovat v obchodě AppExchange.

Klíčová slova GDPR, Salesforce, správa žádostí, osobní údaje

Abstract

This diploma thesis deals with the analysis and the implementation of GDPR Request Management System on the platform Salesforce. The system is used for handling and processing requests which can be put in by the citizens of the European Union according to the *General Data Protection Regulation* (GDPR) issued by the European Parliament.

The system itself was developed using standard tools that the Salesforce platform provides. Furthermore, the GDPR Request Management System was designed so that it can be published in AppExchange store in the future.

Keywords GDPR, Salesforce, request management, personal data

Obsah

Seznam obrázků

Úvod

Informační systémy jsou dnes součástí téměř každého podniku nebo instituce. Tyto subjekty používají informační systémy pro podporu svých procesů a pro zkvalitnění svých služeb. Primárním účelem takového systému je sběr a zpracování strukturovaných či nestrukturovaných dat. Většina těchto dat přímo souvisí nebo se váže ke konkrétním osobám (např. klientům). Tato data nazýváme osobní údaje.

V posledních letech prudce vzrostlo množství osobních údajů, které jednotlivé společnosti shromažďují o svých klientech, protože si uvědomily, že z obchodního hlediska jsou pro ně velmi cenné. Zároveň mnoho společností díky internetu a rozmachu technologií globalizovalo svůj trh a v současnosti tak zpracovává údaje občanů více států. Vzhledem k tomu bylo potřeba najít způsob, jak sjednotit legislativu problematiky osobních údajů na mezinárodní úrovni. Proto Evropská komise a Rada Evropské unie vydala nařízení GDPR (*Obecné nařízení na ochranu osobních údajů*) [1]. Cílem tohoto nařízení je stanovit jednotný právní rámec ochrany osobních údajů v evropském prostoru. Toto nařízení je závazné nejen pro všechny subjekty, jež vykonávají svou činnost na území Evropské unie, ale také pro subjekty, které své služby občanům Evropské unie poskytují.

Cíl práce

Cílem této práce je prostudovat nové nařízení GDPR a shrnout, jaká práva a povinnosti toto nařízení ukládá občanům a institucím. Na základě získaných znalostí je cílem provést analýzu, návrh a implementaci systému pro středně velké podniky, který bude sloužit k podpoře procesu zpracování žádostí, jež budou moci občané nově podávat k jednotlivým institucím. Tento systém bude implementován jako aplikace pro cloudovou platformu Salesforce za pomoci standardních nástrojů platformy. Aplikaci bude v budoucnu možné publikovat v obchodě aplikací AppExchange.

Struktura práce

První kapitola práce popisuje *Obecné nařízení na ochranu osobních údajů* (GDPR) vydané Evropskou komisí a Radou Evropské unie. Cílem kapitoly je nastínit a shrnout problematiku, která bude řešena v dalších částech této práce. Součástí kapitoly je rešerše existujících řešení zabývajících se problematikou správy a zpracování osobních údajů.

Na základě získaných poznatků z první kapitoly je ve druhé kapitole práce provedena analýza a návrh řešení systému pro podporu procesu zpracování žádostí plynoucích z nařízení GDPR. Tento systém je navržen jako aplikace pro platformu Salesforce. V kapitole jsou popsány požadavky na systém, na jejichž základě byl následně proveden návrh s ohledem na aspekty platformy Salesforce.

Následující kapitola se věnuje samotné implementaci systému GDPR Podatelna. V první části jsou představeny použité technologie zvolené platformy. Zbytek kapitoly popisuje implementaci jednotlivých požadavků. K popisu byly vybrány stěžejní části implementace nebo případy, kdy byly použity atypické nástroje platformy.

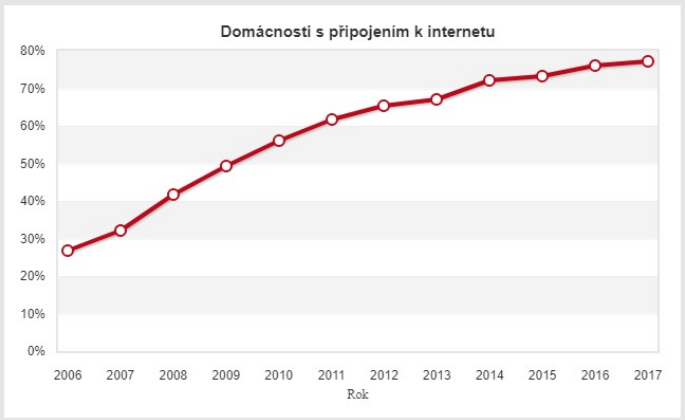
V poslední kapitole je popsána metodika testování implementovaného systému. Testování bylo provedeno pomocí jednotkových a uživatelských testů.

V závěru práce je zhodnocena použitelnost realizovaného systému a jeho možná rozšíření.

Kapitola 1

Obecné nařízení na ochranu osobních údajů

V posledních letech dochází k postupné digitalizaci společnosti. Internet a s ním související technologie se stávají čím dál větší součástí života většiny z nás. Podle Českého statistického úřadu je v České republice připojeno k internetu více než tři čtvrtě domácností (77,2 %) [2] a lze předpokládat, že toto číslo dále poroste (viz graf na obrázku 1.1).



Obrázek 1.1: Domácnosti a připojení k internetu

Jak role internetu a technologií roste, mnoho společností si uvědomuje, že údaje o jejich klientech představují velmi cennou komoditu a jejich zpracováním mohou dosáhnout obchodních úspěchů. Tyto údaje mohou být využívány přímo pro interní činnost společnosti (např. zkvalitnění služeb, marketingu, plánování výroby apod.) nebo s nimi lze obchodovat jako s jakoukoliv jinou komoditou. Údaje tak mohou být například použity pro profilování uživatelů a cílení marketingu na konkrétní skupiny uživatelů.

V posledních letech se stále více řeší, jakým způsobem by se měl digitální svět regulovat a tím hájit zájmy jednotlivých občanů. Posledním známým skandálem, jenž se týkal osobních údajů, byl únik dat 50 milionů uživatelů ze sociální sítě Facebook v březnu 2018. Tato data měla být údajně zneužita společností Cambridge Analytica, která poskytuje služby politickým subjektům, a tím mohlo dojít k ovlivnění voleb v několika zemích [3].

Většina států má svou vlastní právní úpravu, která určitým způsobem reguluje zacházení s osobními údaji občanů. Vzhledem k tomu, že internet je globální a legislativa Evropské unie nereflektovala současný stav, byla přichystána nová legislativní úprava pro ochranu osobních údajů na nadnárodní úrovni.

Obecné nařízení na ochranu osobních údajů neboli GDPR (*General Data Protection Regulation*) je nařízení Evropského parlamentu a Rady Evropské unie, které bylo schváleno 27. dubna 2016 a vstupuje v platnost 25. května 2018 [1]. Toto nařízení přímo stanovuje univerzální pravidla pro zpracování osobních údajů a definuje nově práva subjektům údajů (fyzické osoby). Nařízení je závazné pro všechny členské země Evropské unie. Tyto státy mají možnost nařízení adaptovat vlastní právní úpravou, pomocí které mohou některé části upravit. Nařízení s sebou přináší rovnocennou vymahatelnost práva v celé Evropské unii a usnadní spolupráci dozorových orgánů jednotlivých států.

1.1 Stávající právní úprava v České republice

Právní úprava zacházení s osobními údaji v České republice je dodnes definována především pomocí *zákona č. 101/2000 Sb., o ochraně osobních údajů* [4]. Tento zákon zřizuje *Úřad pro ochranu osobních údajů* (ÚOOÚ), který je nezávislým orgánem, jenž provádí dozor nad dodržováním zákonem stanovených povinností při zpracování osobních údajů.

Adaptační úprava pro GDPR nebude pravděpodobně v České republice do data účinnosti vládou schválena [5] a tak nová pravidla budou přijata přímo pomocí GDPR, jelikož se nejedná o směrnici, ale o nařízení.

Jelikož nařízení GDPR stanovuje nová práva a povinnosti,nahrazuje tak v tomto rozsahu *zákon č. 101/2000 sb. o ochraně osobních údajů*, který bude účinností adaptačního zákona zrušen.

Problematikou správy osobních údajů se zabývá celá řada dalších zákonů a norem.

Mezinárodní organizace pro normalizaci (ISO) vydala mezinárodně platnou normu ISO 27001. Tato norma poskytuje model pro zavedení efektivního řízení bezpečnosti informací v organizaci [6]. ISO 27001 doplňuje normu ISO 27002, jež poskytuje podrobný přehled bezpečnostních opatření pro zajištění bezpečnosti informací.

Dalším zákonem upravujícím legislativu České republiky je Zákon č. 181/2014 Sb., *Zákon o kybernetické bezpečnosti* [7]. Tento zákon upravuje práva a povin-

nosti osob a orgánů veřejné moci v oblasti kybernetické bezpečnosti.

1.2 Nově definované role a pojmy

Nařízení GDPR nově definuje několik nových rolí a pojmů. Tyto definice jsou obsaženy v článku 4 nařízení GDPR [1]. V této kapitole jsou role a pojmy, jež jsou potřeba pro seznámení s problematikou v následujících kapitolách.

1.2.1 Osobní údaj

Osobní údaje jsou ve směrnici definovány jako veškeré informace o identifikované nebo identifikovatelné osobě. Identifikovatelná osoba je fyzická osoba, kterou lze přímo nebo nepřímo identifikovat pomocí osobních údajů. Osobními údaji jsou například jméno, příjmení, identifikátor v rámci IS, kontaktní údaje, lokační údaje nebo například údaje o chování osob (historie objednávek, historie vyhledávání apod.)

Z působnosti nařízení jsou vyjmuty osobní údaje zemřelých osob, anonymizované údaje nebo údaje, které nemají obchodní či institucionální charakter.

Citlivé osobní údaje

Směrnice definuje zvláštní kategorii citlivých osobních údajů, na jejichž zpracování jsou kladeny větší nároky.

Mezi citlivé osobní údaje patří například údaje o rasovém a etnickém původu, politických názorech, náboženském vyznání, sexuálním stavu, zdravotním stavu osoby nebo údaje obsahující genetické a biometrické informace (fotografie, video, otisk prstu apod.). Do této skupiny také spadají veškeré osobní údaje dětí.

1.2.2 Zpracování osobních údajů

Zpracováním osobních údajů je jakákoli automatizovaná či neautomatizovaná činnost, při které dochází ke shromažďování, zaznamenávání, uspořádání, strukturování, ukládání, vyhledávání, použití, šíření nebo zpřístupnění osobních údajů subjektu.

1.2.3 Subjekt údajů

Subjekt údajů je jakákoliv fyzická osoba, ke které se váží osobní údaje. Subjektem není právnická osoba, a tak nelze považovat údaje o právnické osobě za osobní. Výjimkou mohou být údaje o zaměstnancích společnosti (např. kontaktní údaje). Tyto údaje mohou být považovány za osobní údaje, jelikož se vztahují ke konkrétní fyzické osobě.

1.2.4 Správce

Správce je fyzická nebo právnická osoba, instituce či subjekt, který získává osobní údaje o subjektech. Správce určuje, za jakým účelem jsou údaje sbírány, a definuje, jakým způsobem budou data zpracována. Správce primárně zodpovídá za správné nakládání s osobními údaji a za dodržení nařízení GDPR.

1.2.5 Zpracovatel

Zpracovatel je fyzická nebo právnická osoba, instituce či subjekt, kterého si správce najímá, aby pro něj prováděl zpracování osobních údajů. Zpracovatel může provádět pouze takové operace, kterými ho správce pověří nebo vyplývají z činností, k nimž byl zpracovatel pověřen. Za zpracovatele se nepovažuje interní zaměstnanec správce (např. účetní nebo personalista), ani vnitřní útvar společnosti.

Typickým zpracovatelem může být například subdodavatel společnosti (např. účetní firma).

1.2.6 Souhlas subjektu údajů

Souhlas subjektu údajů je jakýkoli svobodný, konkrétní, informovaný a jednoznačný projev vůle, kterým subjekt údajů dává prohlášením či jiným zjevným potvrzením své svolení ke zpracování svých osobních údajů. Tento souhlas opravňuje správce osobních údajů ke zpracování a nakládání s údaji, pokud ke zpracování nemá jiný právní titul.

1.2.7 Dozorový úřad

Dozorový úřad je nezávislý orgán veřejné moci, zřízený vládou jednotlivých členských států. Cílem tohoto úřadu je dohlížet na dodržování nařízení a správné nakládání s osobními údaji občanů. V České republice tuto roli zastává *Úřad pro ochranu osobních údajů* (ÚOOÚ).

1.2.8 Porušení zabezpečení osobních údajů

Za porušení zabezpečení osobních údajů se považuje jednání, které vede k neoprávněnému nakládání s osobními údaji občanů. Jako porušení lze například klasifikovat neoprávněné poskytnutí údajů třetím stranám, jejich zcizení nebo zničení a ztrátu dat.

1.2.9 Pověřenec pro ochranu osobních údajů (DPO)

Podle článku 37 nařízení GDPR [1] má správce nebo zpracovatel povinnost jmenovat *Pověřence pro ochranu osobních údajů* neboli *Data Protection Officer* (DPO) v případě, že:

- zpracování provádí orgán veřejné moci nebo veřejný subjekt
- hlavní činnosti zpracovatele spočívají v rozsáhlém a pravidelném zpracování osobních údajů (např. společnosti podnikající v oboru marketingu)
- hlavní činnosti zpracovatele zahrnují zpracování zvláštních kategorií osobních údajů (citlivé údaje)

Hlavním úkolem DPO je monitorovat soulad zpracování osobních údajů a nařízení GDPR. Pověřenci nenesou osobní odpovědnost za nedodržení GDPR. Za soulad s nařízením jsou zodpovědní správci a zpracovatelé.

Pověřenec musí být jmenován na základě svých profesních kvalit (znalost práva v oblasti ochrany osobních údajů), ale není u něj požadována konkrétní úroveň vzdělání.

1.3 Zásady zpracování

Nařízení GDPR také definuje sadu zásad pro zpracování osobních údajů. Tyto zásady mají zakotvení v článku 5 nařízení [1]. V této části je shrnuto, jakým způsobem by měli správci a zpracovatelé zacházet se všemi osobními údaji.

- Správce má povinnost zpracovávat osobní údaje vůči subjektu transparentně a korektně.
- Správce může osobní údaje zpracovávat, pokud má pro toto zpracování alespoň jeden právní důvod (viz kapitola 1.3.1).
- Správce shromažďuje osobní údaje za určitým a legitimním účelem. Údaje nesmí být zpracovány způsobem neslučitelným s účelem, s nimž byly správci poskytnuty.
- Správce musí provádět tzv. minimalizaci údajů. Evidované údaje musí být přiměřené a relevantní ve vztahu k účelu zpracování.
- Zpracovávané údaje musí být přesné. Správce má povinnost udržovat evidované údaje přesné a v případě potřeby je aktualizovat nebo je případně smazat.
- Údaje subjektu musí být uloženy v přiměřené formě na přiměřeně dlouhou dobu, která je dána účelem zpracování. Údaje mohou být uloženy na delší dobu, pokud je účel jejich archivace ve veřejném zájmu nebo pro účely výzkumu či statistiky.
- Správce má povinnost zajistit náležité zabezpečení osobních údajů pomocí vhodných technických a organizačních opatření, aby zabránil neoprávněnému zpracování údajů, jejich ztrátě nebo poškození.

Tyto pravidla jsou pro správce zásadní. Nejen proto, že podle nařízení mají povinnost je zajistit, ale zároveň také musí být schopni subjektům doložit správné zacházení s osobními údaji a soulad s těmito pravidly. K prokázání souladu mohou sloužit například záznamy o činnostech zpracování.

1.3.1 Právní důvod zpracování

Právní důvod zpracování osobních údajů určuje, jaké má správce oprávnění pro evidování těchto údajů. Pro každý osobní údaj, který je zpracováván, musí mít správce alespoň jeden právní důvod [8]. Pokud by správce zpracovával údaje, ke kterým nemá právní důvod, dopouštěl by se neoprávněného zpracování osobních údajů a tím by porušil nařízení GDPR. Správce by měl zajistit, že údaje bez právního důvodu nebudou zpracovány (např. smazáním).

Jeden konkrétní osobní údaj nebo sada osobních údajů může mít více právních důvodů a účelů zpracování. Například správce může evidovat adresu bydliště subjektu pro potřeby plnění smlouvy (např. doručení dodaného zboží). Zároveň může mít správce od subjektu souhlas s použitím jeho adresy pro potřeby marketingu. Tyto právní důvody můžou v čase vznikat a zanikat. Správce by ale měl vždy zajistit smazání těchto údajů, v případě že pomine poslední právní důvod pro zpracování určitých osobní údajů.

Správce má právo zpracovávat osobní údaje subjektu, pokud je splněn alespoň jeden z následujících 6 právních důvodů:

1. Subjekt udělil souhlas se zpracováním svých osobních údajů pro konkrétní účel.
   - Náležitosti souhlasu jsou popsány v následující kapitole 1.3.2.
2. Zpracování je nezbytné pro plnění smlouvy uzavřené se subjektem nebo pro provedení opatření, která vedou k uzavření smlouvy.
   - např.: evidence kontaktních údajů pro plnění dodávky zboží
3. Zpracování je nezbytné pro splnění právní povinnosti, která je uložena správci.
   - např.: evidence přijatých a vydaných faktur pro potřeby účetnictví
4. Zpracování je nezbytné pro ochranu životně důležitých zájmů subjektu nebo jiné fyzické osoby.
   - např.: evidence zdravotních údajů v lékařských zařízeních
5. Zpracování je nezbytné pro splnění úkolu, který je ve veřejném zájmu.
   - např.: soukromé společnosti, která poskytují služby široké veřejnosti (dodavatelé elektřiny apod.)

6. Zpracování je nezbytné pro účely vykonání oprávněného zájmu zpracovatele.

    - nejflexibilnější z právních titulů
    - např.: aplikační logování informačních systémů pro potřeby IT podpory

1.3.2 Souhlas se zpracováním údajů

Souhlas subjektu se zpracováním jeho osobních údajů je svobodný, konkrétní, informovaný a jednoznačný projev vůle, kterým subjekt dává svolení ke zpracování konkrétních osobních údajů. Aby byl souhlas v souladu s článkem 7 nařízení GDPR [1], musí splňovat následující požadavky:

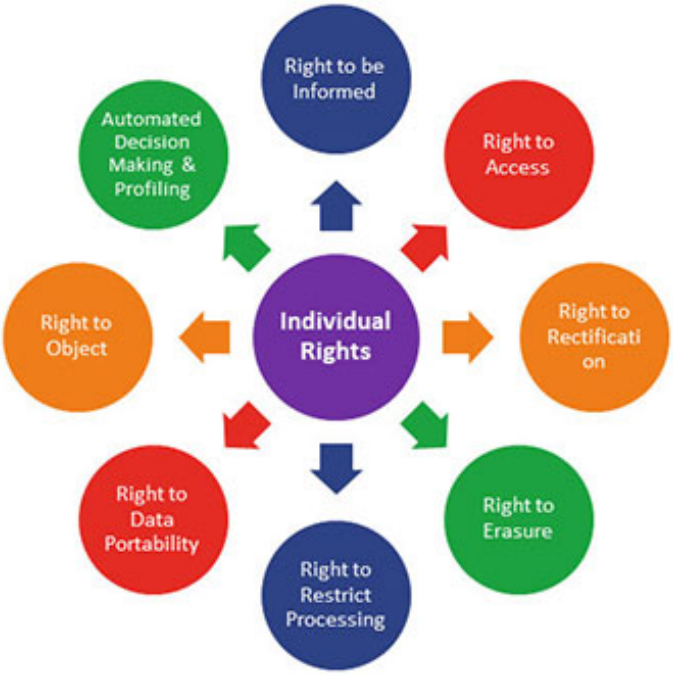
- Souhlas se poskytuje vždy k určitému účelů a tento účel musí subjekt při vyslovení souhlasu znát.
- Subjekt má právo souhlas odvolat. Odvoláním souhlasu nemusí vždy dojít k smazání dat správcem. Data jsou odvoláním souhlasu smazána pouze v případě, že tento souhlas byl jediným právním důvodem zpracování.
- Souhlas není potřeba pro zpracování, jehož účel má jiné právní důvody.
- Pokud je zpracování osobních údajů založeno na základě souhlasu, má správce povinnost tyto souhlasy evidovat a případně je doložit.
- V případě, že je souhlas vyjádřen písemným prohlášením, které se zároveň týká jiných skutečností, musí být popis souhlasu jasně oddělen a musí být napsán za použití jasných a jednoduchých jazykových prostředků. Souhlas by například neměl být skryt v textu všeobecných podmínek.
- V případě poskytování služeb nezletilým osobám, má správce povinnost u osob mladších 16 let získat zároveň souhlas osoby, která vykonává rodičovskou zodpovědnost dítěte (zákonný zástupce). Jednotlivé členské státy EU si mohou tuto hranici upravit. Hranice však nesmí být nižší než 13 let. Správce má povinnost vyvinout přiměřené úsilí s ohledem na dostupné technologie, aby ověřil platnost souhlasu zákonného zástupce.

Nařízení GDPR umožňuje použití souhlasů, jež správce získal od subjektů před platností nařízení. Podmínkou je, aby tyto souhlasy byly v souladu s nařízením. Vzhledem k tomu, že dřívější souhlasy například nemusí obsahovat všechny potřebné informace (např. výčet všech zpracovatelů) nebo nebyly uzavřeny validním způsobem (např. byly součástí všeobecných podmínek), budou muset správci tyto souhlasy obnovit ve znění, které je v souladu s nařízením.

1.4 Práva subjektů

Jedním z největších dopadů nařízení GDPR na instituce je výrazné posílení práv subjektů. Hlavní motivací tohoto opatření je snaha vybalancovat nerovnoměrný vztah mezi správcem osobních údajů a subjekty.

Nařízení nově definuje sadu žádostí, které mohou subjekty podávat k správcům svých osobních údajů. Subjektu je pomocí těchto žádostí umožněno, aby získal přehled o zpracování svých osobních údajů, a tím je naplněna zásada transparentnosti zpracování. Přehled typů žádostí je zobrazen na obrázku 1.2.



Obrázek 1.2: Sada práv subjektu

Forma podání žádosti není striktně definována. Nařízení umožňuje podat žádost osobně u správce, elektronicky přes internet nebo jinými vhodnými prostředky. Tato forma by měla odpovídat způsobu sběru osobních údajů.

Správce je povinen zajistit vhodnou metodiku pro identifikaci subjektů. Tato metodika bude silně závislá na typu instituce, která data zpracovává. Pokud má správce důvodné pochybnosti o totožnosti subjektu, může subjekt požádat o poskytnutí dodatečných informací nezbytných k potvrzení totožnosti subjektu.

Správce má povinnost vyhovět žádosti subjektu do jednoho měsíce od data obdržení žádosti. Tato lhůta může být s ohledem na složitost žádosti prodloužena až na 3 měsíce. V takovém případě je ovšem správce povinen subjekt o prodloužení informovat, společně s odůvodněním odkladu.

Jestliže správce není schopen žádosti vyhovět, je povinen subjekt bezodkladně informovat o důvodech nevyhovění a o možnosti podat stížnost k dozorovému úřadu.

Zpracování žádostí musí být pro subjekty bezplatné. Pokud by subjekt zneužíval bezplatnosti a podával by zjevně nedůvodné a nepřiměřené žádosti, má správce právo

1. uložit přiměřený poplatek, který zohledňuje administrativní náklady spojené se zpracováním požadavku

2. žádost zamítnout

V případě těchto opatření má správce povinnost své konání subjektu zdůvodnit.

1.4.1 Právo na informace

Subjekt má podle článku 12 nařízení GDPR [1] právo na to být informován (*Right to be informed*), jakým způsobem jsou zpracovány jeho osobní údaje.

Toto právo slouží především k získání informací o účelu zpracování osobních údajů, totožnosti správce, jeho oprávněných zájmech a o poskytování dat jiným zpracovatelům.

U tohoto jediného práva, by měl prvotní aktivitu provést správce a poskytnout informace všem subjektům.

V případě žádosti na toto právo ze strany subjektu, může být odpovězeno například odkázáním na informační memorandum na internetových stránkách nebo jeho odeslání emailem.

1.4.2 Právo na přístup k osobním údajům

Subjekt má podle článku 15 nařízení GDPR [1] právo na přístup ke svým osobním údajům, které správce zpracovává (*Right of access*). Subjekt má právo být informován i v případě, že u správce nedochází ke zpracování jeho údajů.

Součástí odpovědi by pro osobní údaje měly být zahrnuty tyto údaje:

- účel zpracování

- kategorie osobních údajů

- příjemci, kterým jsou osobní údaje zpřístupněny, zejména jedná-li se o příjemce ve třetích zemích

- informace, zda je možné od správce požadovat opravu, výmaz nebo omezení zpracování

- právo podat stížnost u dozorového orgánu

- informace o zdroji údajů, nejedná-li se o údaje získané od subjektu

- informace, zda dochází pomocí těchto údajů k automatizovanému rozhodování (např. profilování)

Správce musí subjektu poskytnout kopii zpracovávaných osobních údajů. Jestliže byla žádost podána elektronicky, poskytnou se údaje v elektronické formě.

Plněním tohoto práva nesmí dojít k porušení práva a svobody jiných subjektů. Správce tudíž musí zajistit, že subjektu předá pouze správné údaje.

1.4.3 Právo na opravu

Podle článku 16 nařízení GDPR [1] má subjekt právo vznést žádost na opravu svých osobních údajů (*Right to rectification*), u kterých se domnívá, že by mohly být nepřesné.

Správce by měl zajistit, že tyto žádosti lze podat elektronicky, především v případě, že zpracování osobních údajů probíhá přes internet.

1.4.4 Právo na výmaz

Článek 16 nařízení GDPR [1] dává subjektům právo žádat správce o vymazání osobních údajů (*Right to erasure*). Právo je také označováno jako „právo být zapomenut“.

Správce by měl bez zbytečného odkladu vymazat osobní údaje, pokud je dán jeden z těchto důvodů:

- Osobní údaje již nejsou potřebné pro účel, pro který byly zpracovány.
- Subjekt odebere souhlas a neexistuje jiný právní důvod ke zpracování.
- Subjekt vznese námitku proti zpracování a neexistují žádné jiné oprávněné důvody ke zpracování.
- Osobní údaje byly zpracovány protiprávně.
- Neexistuje rodičovský souhlas pro zpracování údajů nezletilých.

1.4.5 Právo na omezení zpracování

Subjekt má na základě článku 18 nařízení GDPR [1] právo požadovat po správci, aby omezil zpracování osobních údajů subjektu (*Right to restrict processing*).

Správce má omezit zpracování osobních údajů subjektu v těchto případech:

- Subjekt označuje osobní údaje za nepřesné a po dobu vyřízení je potřeba omezit jejich zpracování.
- Zpracování osobních údajů je protiprávní a subjekt odmítá jejich výmaz.

- Správce údaje nepotřebuje pro potřeby zpracování, ale subjekt požaduje jejich uložení například pro obhajobu právních nároků.
- Subjekt vznesl námitku proti zpracování a je potřeba omezit zpracování po dobu než bude námitka vyřešena.

V případě, že jsou osobní údaje zpracovány automaticky (např. pomocí IS), je potřeba zajistit, aby tyto údaje nebyly dále neoprávněně zpracovány.

1.4.6 Právo na přenositelnost údajů

Subjekt má podle článku 20 nařízení GDPR [1] právo získat od správce poskytnutá data ve strukturované podobě v běžně používaném a strojově čitelném formátu (*Right to data portability*). Tyto údaje může předat jinému správci a původní správce údajů mu v tom nemůže bránit.

Toto právo může být požadováno, pokud jsou splněny tyto podmínky:

- Osobní údaje subjektu jsou zpracovávány na základě souhlasu nebo plnění smlouvy.
- Zpracování údajů se provádí automatizovaně.

1.4.7 Právo vznést námitku

Článek 21 nařízení GDPR [1] poskytuje subjektům právo vznést námitku proti zpracování osobních údajů (*Right to object*).

Tato žádost je podána v případě, že subjekt nemá možnost uplatnit právo na výmaz osobních údajů a nepřeje si zpracování údajů za konkrétním účelem.

Taková situace může nastat například v případě, že správce zpracovává údaje za účelem splnění úkolu ve vyšším zájmu a subjekt si nepřeje údaje zpracovávat jiným způsobem.

1.4.8 Automatizované individuální rozhodování

Subjekt má podle článku 22 nařízení GDPR [1] právo nebýt předmětem žádného rozhodnutí založeného výhradně na automatizovaném zpracování. Toto právo se týká také profilování, které se subjektu dotýká.

Toto právo nelze uplatnit v následujících případech:

- Rozhodnutí je nezbytné pro uzavření nebo plnění smlouvy mezi správcem a subjektem.
- Rozhodnutí je povoleno právem členského státu nebo EU.
- Pro rozhodování má správce od subjektu souhlas.

1.5 Povinnosti správců

Součástí nového nařízení GDPR je také sada povinností, jež nově náleží správcům osobních údajů subjektů.

Hlavní povinností správce je zajistit správné a legitimní zpracování osobních údajů v souladu s nařízením GDPR. Konkrétní podoba potřebných opatření správce je silně závislá na typu jeho činnosti a také na tom, jaké osobní údaje zpracovává a jakým způsobem je zpracovává.

Správce by měl vyvinout přiměřené úsilí k zajištění souladu s nařízením. V tomto ohledu je nařízení celkem volné a bude vždy záležet na konkrétní situaci správce. Největší pozornost by měly nařízení věnovat společnosti zabývající se primárně zpracováním osobních údajů (např. marketing), společnosti pracující s citlivými osobními údaji (např. lékařská zařízení) nebo společnosti, jež sbírají větší množství osobních dat než je běžné.

V této sekci jsou shrnuty jednotlivé povinnosti, které má nově správce. Nakonec jsou popsány možné sankce za nedodržení povinností.

1.5.1 Povinnost posouzení vlivu na ochranu osobních údajů

Správce by měl ještě před samotným zahájením zpracování provést posouzení, jaký vliv bude mít zpracování na osobní údaje subjektů. Toto posouzení je potřeba udělat zejména v případě, že zpracování provádí rizikové operace s osobními údaji a data mohou být potenciálně zneužita.

Posouzení vlivu by mělo obsahovat systematický popis zamýšleného zpracování a posouzení rizik z hlediska práv subjektů.

Úřad pro ochranu osobních údajů připravuje seznam operací zpracování, kdy bude nutné provést toto posouzení vlivu.

1.5.2 Povinnost vést záznamy o zpracování

Správce a zpracovatelé mají povinnost vést záznamy o činnostech zpracování, za které zodpovídají [9].

Této povinnosti jsou zproštěny organizace, jež mají méně než 250 zaměstnanců, jejich hlavní činností není zpracování osobních údajů a nezpracovávají citlivé osobní údaje subjektů.

Záznamy o činnosti by měly obsahovat tyto údaje:

- jméno a kontaktní údaje správce a zpracovatele
- údaje pověřence pro ochranu osobních údajů
- popis kategorií subjektů a kategorií jejich osobních údajů
- kategorie příjemců se kterými jsou údaje sdíleny
- informace o mezinárodním předání informací

- lhůty pro výmaz jednotlivých údajů
- popis technických a organizačních opatření

1.5.3 Povinnost spolupracovat s dozorovým úřadem

Každý správce a zpracovatel má povinnost na vyžádání spolupracovat s dozorovým úřadem (*Úřad pro ochranu osobních údajů*). Tomuto úřadu budou muset doložit, jakým způsobem zpracovávají osobní údaje, a také budou muset zpřístupnit záznamy o zpracování.

1.5.4 Povinnost hlásit porušení

Správce má povinnost v případě incidentu, který vede k neoprávněnému zpracování osobních údajů subjektů, podat hlášení dozorovému úřadu. Tato povinnost nastává především pokud porušení zabezpečení představuje vysoké riziko pro práva a svobody subjektů.

Toto hlášení by mělo být podáno bez odkladu a pokud možno do 72 hodin od okamžiku, kdy se o něm správce dozvěděl.

1.5.5 Povinnost jmenovat pověřence pro ochranu

Správce má za určitých podmínek povinnost jmenovat pověřence pro ochranu osobních údajů (DPO). Více informací o pověřenci a podmínky jeho jmenování jsou popsaný v kapitole 1.2.9.

1.5.6 Další povinnosti

Kromě výše uvedených povinností, má správce řadu dalších, je jsou přímo definovány právy subjektů. Tato práva jsou popsána v předchozí kapitole 1.4.

1.5.7 Sankce za porušení povinností

Článek 83 nařízení GDPR [5] umožňuje dozorovému úřadu, aby v případě porušení nařízení udělil správcům údajů správní pokutu. Úřad by měl zajistit, že tyto pokuty budou účinné, přiměřené a odrazující.

Výše správní pokuty je závislá na okolnostech konkrétního případu a závisí na následujících faktorech:

- povaha, závažnost a délka porušení
- počet dotčených subjektů a míra škody
- zda k porušení došlo úmyslně nebo z nedbalosti
- učiněné kroky správce ke zmírnění škod

- míra zodpovědnosti správce s přihlédnutím k technickým a organizačním opatřením
- předchozí porušení správce
- spolupráce správce s dozorovým úřadem
- dotčené kategorie osobních údajů
- způsob oznámení dozorovému úřadu
- jiné přitěžující nebo polehčující okolnosti

Dozorový úřad má možnost za nesplnění příkazu udělit instituci pokutu ve výši, která je určena vyšší hodnotou z těchto dvou částek:

- 4 % z celkového celosvětového ročního obratu instituce
- 20 milionů EUR

Pokud správce nebo zpracovatel poruší více ustanovení z nařízení, nesmí celková výše pokuty překročit výši stanovenou pro nejzávažnější porušení.

1.6 Dostupná řešení

Jak již bylo zmíněno v předchozí kapitole 1.5, konkrétní implementace GDPR ve společnostech závisí na mnoha faktorech.

Důležité je zmínit, že správce by měl vyvinout přiměřené úsilí k zajištění souladu s nařízením. Z toho důvodu nelze obecně říci, jaké kroky by instituce měly podniknout, aby řádně splnily povinnosti plynoucí z nařízení GDPR.

Každá instituce by si měla v první řadě provést interní audit osobních údajů a stanovit, jaké údaje subjektů shromažďuje, za jakým účelem a z jakého právního důvodu. Na základě těchto informací by měly být podniknuty další kroky.

1.6.1 Konzultantské služby

S blížícím se datem účinnosti nařízení se rozrostl počet obchodních společností, které nabízí konzultantské služby v oboru ochrany osobních údajů a pomáhají institucím s kroky, jež vedou k souladu s nařízení GDPR.

V další části této sekce jsou uvedeny některé z těchto společností s popisem jejich činnosti.

O2 GDPR Partner

Společnost O_2 poskytuje nástroj *O2 GDPR Partner*, který pomáhá odhalit rizika při zpracování osobních dat z hlediska nařízení GDPR [10].

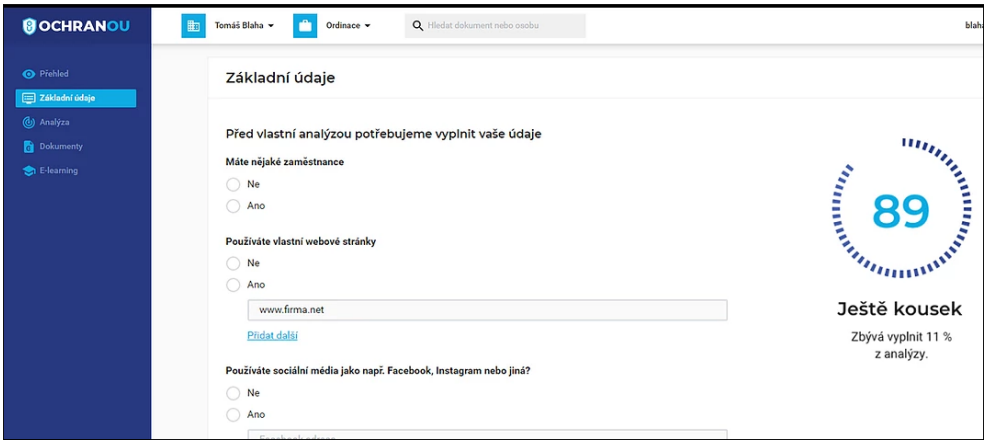
- forma online dotazníku
- série kvízových otázek (u složitějších otázek je poskytnuta nápověda a vysvětlení)
- výstupem sada individuálních doporučení, jak se dále připravit na GDPR
- cena: 1500 Kč bez DPH

Tato služba je určena především pro společnosti, u nichž lze předpokládat minimální dopad nařízení.

OCHRANOU

Aplikace OCHRANOU je produkt, který byl vyvinut Centrem pro ochranu osobních údajů [11]. Tato aplikace slouží k usnadnění implementace GDPR formou dotazníku [12].

- komplexní online dotazník v aplikaci
- webová aplikace - funkční na všech zařízeních
- výsledkem vytvořené dokumenty v souladu s GDPR
- cena: 16 990 - 18 990 Kč bez DPH (podle typu balíčku)
- možnost platby podpory za měsíční poplatek (490 Kč bez DPH)



Obrázek 1.3: Ukázka aplikace OCHRANOU

Aplikace je nabízena ve 3 různých variantách:

- Ordinace – pro kliniky a jiná lékařská zařízení
- Malé a střední firmy – pro firmy a podnikatele do 50 zaměstnanců
- Velké firmy – řešení na míru

Konzultace na míru

Na trhu je řada dalších společností, které nabízí právní poradenství v oblasti ochrany osobních údajů a nařízení GDPR. Tyto konzultace nejsou ovšem nijak automatizované a jejich provedení je individuální. Pomocí takové konzultace vznikne pravděpodobně kvalitnější implementace nařízení, cena těchto konzultací však bude o řád vyšší než u výše uvedených řešení.

1.6.2 Software pro GDPR

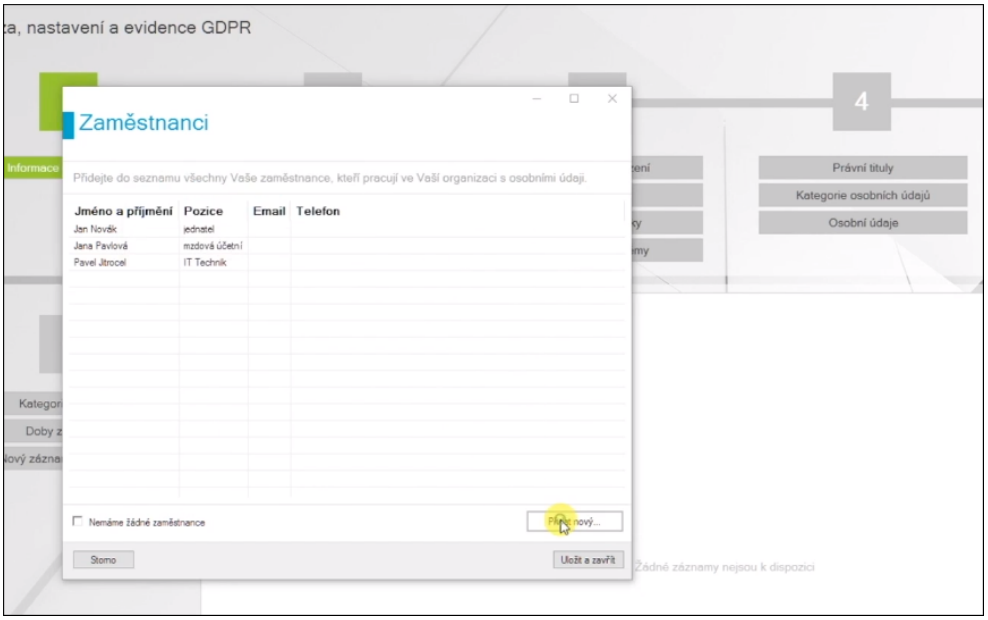
Produkty popsané v předchozí kapitole 1.6.1 poskytují doporučení nebo vygenerované dokumenty pouze na základě vyplněných dotazníků. V této části jsou popsány softwarová řešení, která usnadňují konkrétní procesy vykonávání souladu s nařízením GDPR.

xGDPR Express

Aplikace xGDPR Express slouží k zavedení GDPR krok za krokem a je určena malým a středním firmám, školám, obcím, neziskovým organizacím a podobným menším subjektům [13]. Zavedení aplikace probíhá opět formou průvodce, pomocí kterého jsou data sbírána. Dále systém umožňuje správu agendy, která souvisí s následným plněním nařízení.

Toto řešení nabízí následující funkcionalitu:

- evidence právních titulů zpracovávaných osobních údajů
- stanovení účelu a doby zpracování
- evidence zpracovatelů a příjemců
- generování prohlášení a souhlasů
- generování tiskových výstupů na žádosti subjektů
- evidence školení
- evidence bezpečnostních incidentů
- evidence námitek, žádostí na opravu a žádostí o výmaz



Obrázek 1.4: Ukázka aplikace xGDPR Express

Výrobce softwaru připravil společně s produktem sadu školení, které uživatele seznamují s nařízením GDPR a dávají návod, jak s aplikací zacházet.

Systém je distribuován jako krabicový software, který je provozován na počítači zákazníka.

Cena tohoto řešení je 14 990 Kč bez DPH za jednu licenci, která postačuje pro celou organizaci.

eDPO

Systém eDPO je SW nástroj pro zavedení a provoz GDPR od společnosti DataLite [14]. Tento systém je určen pro celou řadu typů institucí - organizace, holdingy, pověřence, lékařské subjekty, školy či pro státní a veřejné instituce. Systém je nabízen jako cloudová služba a klientovi k použití postačuje standardní počítač a webový prohlížeč.

Systém podporuje následující funkcionalitu:

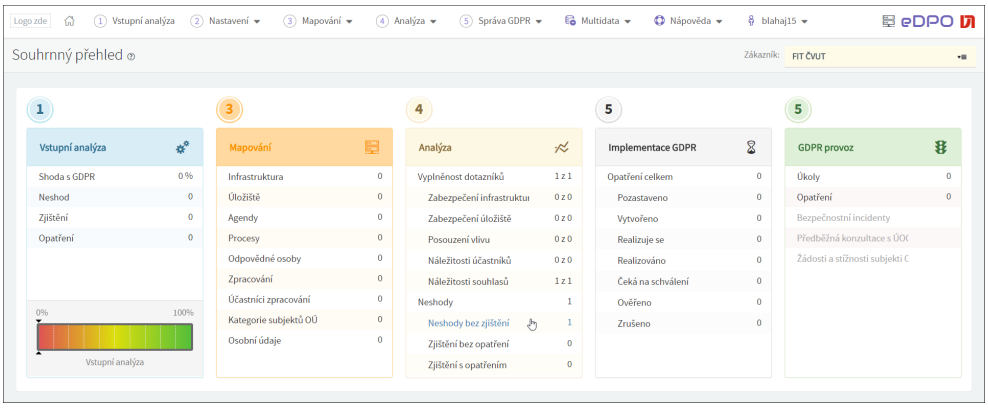
- vstupní analýzy GDPR
- průvodce implementací GDPR pomocí dotazníků
- analýza zabezpečení osobních údajů
- šablony standardních agend
- správa souhlasů subjektů

- evidence zpracovatelů a jejich smluv
- analýza rizik
- evidence a posouzení zjištění
- zpracování a export statistik
- přijímání a evidence žádostí
- tisk a export dokumentace
- správa metodických pokynů
- evidence komunikace s dozorovým úřadem
- neomezený počet uživatelů v rámci organizace

Systém je nabízen ve dvou variantách:

- eDPO Single - určeno pro jednu organizaci
- eDPO Multi - určeno pro skupinu organizací, které jsou spravovány jedním pověřencem pro ochranu osobních údajů

Cena tohoto řešení startuje na 650 Kč bez DPH za měsíc (základní verze) [15].

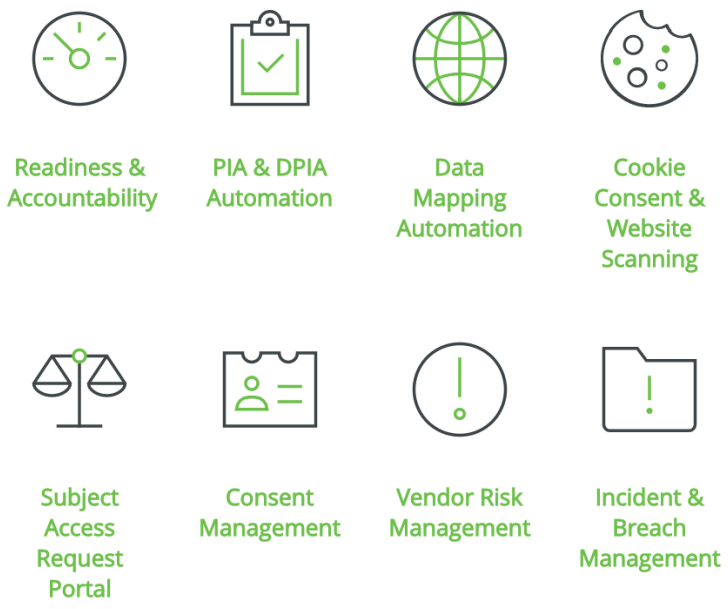


Obrázek 1.5: Ukázka aplikace eDPO

OneTrust

Společnost OneTrust a její produkty jsou globálně nejrozšířenější řešení pro správu osobních údajů [16]. Toto řešení je určeno především pro velké nadnárodní společnosti, které spravují velké množství dat svých o zaměstnancích nebo klientech.

Toto řešení se skládá z jednotlivých modulů, které mají za cíl řešit jednotlivé okruhy problematiky správy osobních údajů. Přehled modulů je zobrazen na obrázku 1.6.



Obrázek 1.6: Přehled modulů systému OneTrust

Jedním z těchto modulů je *Subject Access Request Portal*, který se zabývá evidencí a správou žádostí, jež mohou subjekty podávat ke správcům údajů.

Rozsáhlosti a složitosti problematiky, kterou se systém zabývá, odpovídá také cena tohoto řešení. Systém je zpoplatněn po jednotlivých modulech. Cena modulů se pohybuje v rozmezí 500 - 1 500 USD za měsíc. Podrobnější licenční podmínky jsou popsány na stránce dodavatele [17].

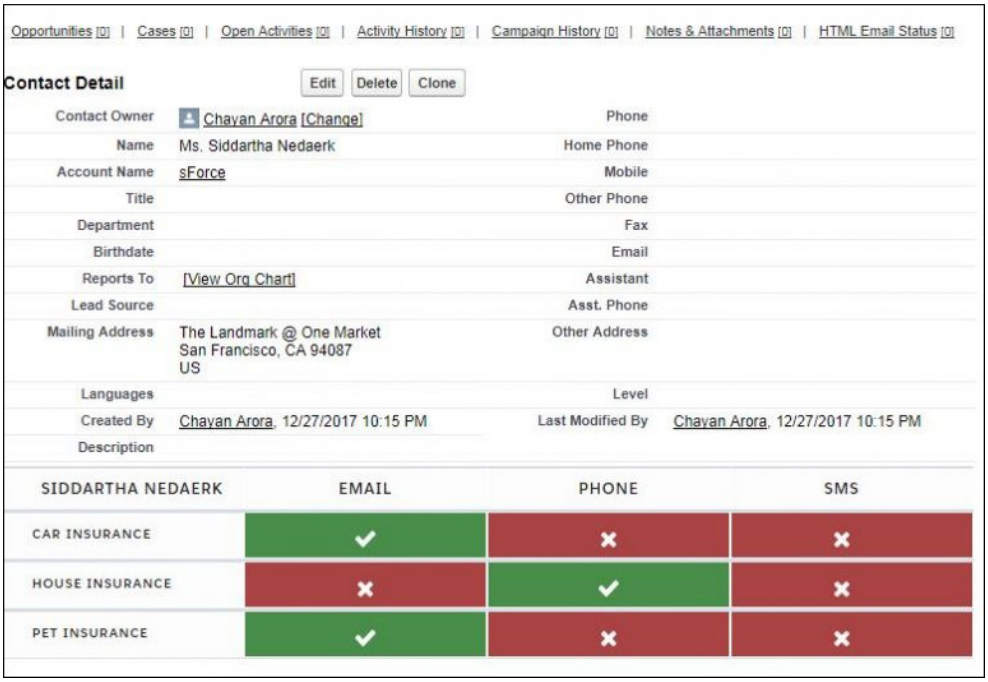
DataPro Tools

DataPro Tools od společnosti Quality System Solutions [18] je jediná aplikace ke správě osobních dat přímo na platformě Salesforce. Aplikace je do Salesforce instalována přes obchod AppExchange jako managed package (popis technologie viz 2.6.3.2).

Řešení nabízí následující funkcionalitu:

- správa právních důvodů pro zpracování údajů
- evidence souhlasů subjektů
- auditní logování pro aplikaci práva na výmaz
- Vyhledávání objektů `Contact` a `Lead` podle typu souhlasu a účelu zpracování
- hromadné mazání a úprava záznamů, pro které neexistuje příslušný právní důvod

Cena tohoto modulu je 10 GBP za jednoho uživatele měsíčně [19].



Obrázek 1.7: Ukázka aplikace DataPro Tools

1.6.3 Shrnutí

Obecné nařízení o ochraně osobních údajů (GDPR) nově dává občanům sadu práv a institucím nové povinnosti. Hlavním cílem nařízení je vyvážit nerovnoměrný vztah mezi občany a institucemi, který je v oblasti zpracování osobních dat.

Při zavádění opatření k zajištění souladu s GDPR by společnosti v první řadě měly provést interní audit jejich zpracování osobních dat klientů, který určí, jaké další kroky budou muset učinit. K tomu mohou využít nabízených služeb společností, jejichž činnost je popsána v kapitole 1.6.1.

V případě že společnost zpracovává osobní údaje ve větším rozsahu a naplnění souladu s nařízením GDPR bude komplikovanější, může společnost využít softwarová řešení, jež jsou popsány v kapitole 1.6.2.

Kapitola 2

Analýza a návrh

V této kapitole je popsáno, jak probíhala analýza, sběr požadavků a návrh systému GDPR Podatelna, jehož hlavním cílem je poskytnout funkcionalitu pro zpracovávání žádostí týkajících se zpracování osobních údajů. Tyto žádosti nově mohou podávat subjekty podle nařízení GDPR. Jednotlivé typy žádostí jsou popsány v kapitole 1.4.

Systém GDPR Podatelna je určen pro středně velké podniky, které mají řadu informačních systémů, pomocí kterých sbírají a zpracovávají osobní údaje klientů pro podporu svých procesů. Největší přínos má systém pro společnosti, jež používají platformu Salesforce nebo plánují její zavedení. Systém není určen pro společnosti, jejichž hlavní činnost je zpracování osobních údajů a jejich přeprodej (reklamní společnosti apod.)

Systém bude vyvinut jako samostatný modul cloudové platformy Salesforce za pomoci standardních vývojových prostředků této platformy. Modul bude navrhnut a naimplementován tak, aby jej bylo možné v budoucnu publikovat v obchodě aplikací AppExchange [20].

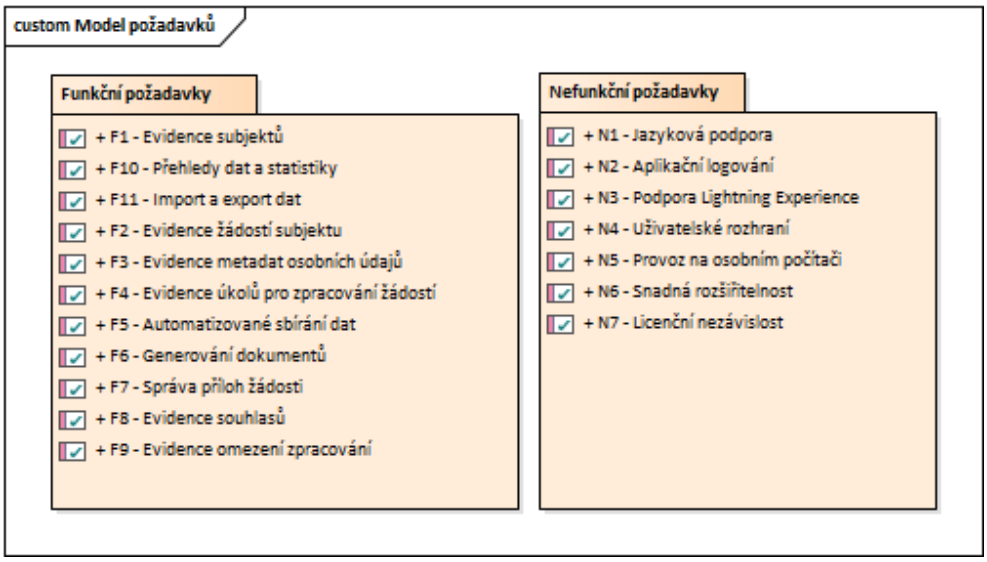
2.1 Analýza požadavků

Na základě rešerše problematiky nařízení GDPR v kapitole 1 byly stanoveny požadavky na výsledný systém. Tyto požadavky definují funkcionalitu systému a vymezují jeho hranice.

Požadavky jsou rozděleny do dvou skupin - funkční a nefunkční. Přehled a rozdělení všech požadavků je znázorněn na diagramu 2.1.

2.1.1 Funkční požadavky

V této sekci jsou popsány funkční požadavky systému. Tyto požadavky definují primární funkčnost a vlastnosti systému.



Obrázek 2.1: Model požadavků

F1 – Evidence subjektů

Systém bude umožňovat evidenci subjektů údajů. Tyto záznamy reprezentují subjekt údajů definovaný nařízením GDPR (definice viz kapitola 1.2.3). Subjekt je hlavní entitou aplikace a všechny ostatní objekty se přímo nebo nepřímo vážou k této entitě.

Klient bude mít možnost rozšířit základní sadu atributů subjektu (jméno, příjmení, kontaktní údaje) o své další vlastní atributy.

F2 – Evidence žádostí subjektu

U subjektů bude možné evidovat jednotlivé žádosti na plnění práv nařízení GDPR. Žádosti budou rozděleny na tyto základní typy:

- Právo na informace
- Právo na přístup k osobním údajům
- Právo na opravu
- Právo na výmaz
- Právo na omezení zpracování
- Právo na přenositelnost údajů
- Právo vznést námitku

- Obecná žádost - jiná žádost subjektu na zpracování osobních údajů

Vysvětlení a příklad jednotlivých žádostí je popsán v první kapitole v sekci 1.4. Pro jednotlivé typy žádostí bude mít klient možnost definovat jiné uživatelské rozhraní a sadu zobrazených atributů.

F3 – Evidence metadat osobních údajů

Systém bude umožňovat definovat metadata osobních údajů. Metadata určují, které osobní údaje jsou zpracovávány v jakých systémech. Dále definují, jakým způsobem lze konkrétní data subjektu získat.

Jednotlivé osobní údaje bude možné spojovat do skupin. Tyto skupiny budou použity například pro generování dokumentů.

F4 – Evidence úkolů pro zpracování žádostí

Systém bude pro *Právo na přístup k osobním údajů* a *Právo na přenositelnost* poskytovat funkcionalitu pro sesbírání dat. Pro jednotlivé typy osobních údajů budou podle metadat vygenerovány úkoly pro získání dat. Úkoly budou přiřazeny konkrétním správcům informačních systémů a budou obsahovat veškeré potřebné informace pro získání dat. Tyto úkoly bude možné zpracovat automaticky nebo manuálně.

F5 – Automatizované sbírání dat

Zpracování úkolů na sbírání dat v rámci platformy Salesforce bude možné provést automaticky. Způsob, jakým mají být data získána, bude mít možnost klient definovat.

Data bude možné sbírat na úrovni jednotlivých atributů (např. telefonní číslo) nebo na úrovni celého záznamu (např. historie objednávek). V druhém případě bude možné definovat, jaké atributy mají být pro záznam získány.

Pro data mimo platformu Salesforce (např. externí systém) bude mít klient možnost doimplementovat automatizované sbírání těchto dat. Toho může být docíleno například pomocí API, které externí systém poskytne.

F6 – Generování dokumentů

Pro jednotlivé žádosti bude možné vygenerovat odpověď ve formátu PDF [21]. Způsob generování dokumentu bude určen na základě typu žádosti. Klient bude mít možnost doimplementovat vlastní logiku pro určení způsobu generování.

Textaci jednotlivých dokumentů bude možné měnit pomocí šablon.

Po zobrazení náhledu bude možné dokument uložit ke konkrétní žádosti subjektu.

F7 – Správa příloh žádosti

K jednotlivým žádostem bude možné nahrávat dokumenty libovolného typu (např. odpovědi na žádost, dokumenty potřebné ke zpracování apod.). Tyto dokumenty bude možné verzovat.

F8 v Evidence souhlasů

U subjektů bude možné evidovat jeho souhlasy se zpracováním. Souhlas se váže ke skupině osobních údajů. Sledované atributy bude možné definovat klientem.

Při generování odpovědi na *Právo na přístup k osobním údajům* jsou tyto souhlasy uvedeny jako právní důvody zpracování.

F9 – Evidence omezení zpracování

V případě, že subjekt aplikuje *Právo na omezení zpracování*, bude možné u subjektu tyto omezení evidovat. Omezení se vztahuje ke skupině osobních údajů a klient bude mít možnost evidované atributy záznamu rozšířit.

F10 – Přehledy dat a statistiky

Systém bude umožňovat tvořit náhledy nad záznamy jednotlivých objektů. Záznamy bude možné filtrovat na základě hodnot záznamů. Dále bude možné definovat, jaké atributy mají být v náhledu zobrazeny.

F11 – Import a export dat

Systém bude pro všechny entity datového modelu umožňovat exportování a importování dat ve formátu CSV [22].

2.1.2 Nefunkční požadavky

Nefunkční požadavky sice nedefinují primární funkce systému, ale přesto je potřeba je zohlednit v návrhu a implementaci výsledného řešení.

N1 – Jazyková podpora

Uživatelské prostředí bude vyvinuto v anglickém jazyce. Systém bude umožňovat jeho snadné přeložení do jiných jazyků (např. čeština) pomocí standardních nástrojů Salesforce.

N2 – Aplikační logování

Systém bude evidovat provedené akce a změny v průběhu zpracování žádostí. Tyto záznamy budou přístupné pouze pro administrátory.

N3 – Podpora Lightning Experience

Systém bude možné používat v novém uživatelském prostředí Salesforce Lightning Experience.

N4 – Uživatelské rozhraní

Vzhled uživatelského rozhraní bude odpovídat standardům platformy Salesforce a pro jeho implementaci budou použity nástroje platformy (SLDS).

N5 – Provoz na osobním počítači

Aplikace bude přístupná přes platformu Salesforce z prohlížeče Google Chrome verze 61 nebo vyšší [23]. Vzhled a stylování aplikace budou optimalizované pro Full HD (1920 x 1080) rozlišení.

N6 – Snadná rozšiřitelnost

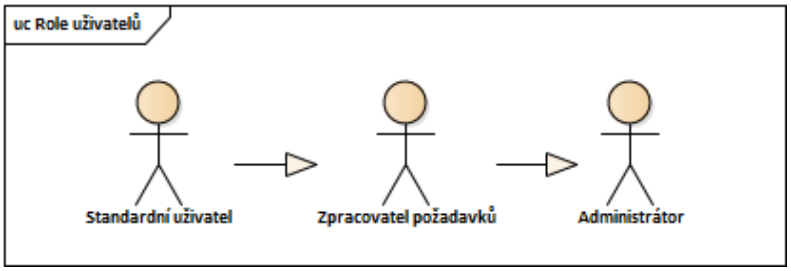
Systém bude vyvinut tak, aby jej bylo snadné v budoucnu rozšířit o další funkcionalitu. Dále je požadována snadná rozšiřitelnost na straně klienta, jenž si aplikaci nainstaluje.

N7 – Licenční nezávislost

Při vývoji budou použity pouze nástroje a knihovny třetích stran, jejichž licenční podmínky umožňují volné použití i pro komerční účely.

2.2 Role uživatelů

Uživatelé systému budou rozděleni do tří skupin. Tyto skupiny určují, jakou sadu oprávnění uživatel bude v systému mít. Všechny tři skupiny jsou hierarchické. Uživatel nadřazené skupiny má alespoň stejná práva jako uživatel z podřazené skupiny. Vztah jednotlivých rolí uživatelů je zobrazen na diagramu 2.2.



Obrázek 2.2: Role uživatelů

2.2.1 Běžný uživatel

Uživatel zodpovídá za přijetí žádosti od subjektu, její založení do systému a následné předání odpovědi zpět subjektu.

Tento uživatel má minimum oprávnění, jelikož samotné zpracování žádosti není v jeho kompetenci. Uživatel má právo na vytvoření a úpravu záznamů objektů `Subjekt` a `Žádost`.

2.2.2 Zpracovatel požadavků

Uživatel z této skupiny provádí samotné zpracování žádostí. Uživatel má veškerá oprávnění potřebná ke zpracování žádosti (přístup k záznamům metadat osobních údajů, funkcionalita pro sbírání dat apod.)

Dále má uživatel právo vytvářet, upravovat a mazat záznamy o zpracování dat subjektu (`Souhlasy`, `Omezení`, `Identifikátory subjektu`).

V této skupině jsou také uživatelé, kteří zodpovídají za manuální zpracování úkolů pro získání dat (např. z externích systémů).

2.2.3 Administrátor

Administrátor provádí instalaci aplikace do instance Salesforce a je zodpovědný za její provoz. Tento uživatel by měl mít zkušenosti s administrací Salesforce a také s vývojem na této platformě.

Uživatel má plné oprávnění k úpravě a mazání záznamů všech objektů. Dále může definovat metadata osobních údajů pro zpracování nebo šablony generovaných odpovědí.

Veškeré klientské úpravy musí provádět uživatel s administrátorským přístupem.

2.3 Entity doménového modelu

V této sekci jsou popsány entity doménového modelu, jejich atributy a vztahy mezi nimi. Tento model byl navržen na základě sběru a analýzy požadavků.

Pro každou entitu bude vytvořen `Custom Object`. Tento objekt slouží k zajištění perzistence. Salesforce nabízí velkou škálu funkcionality, která je přístupná pro všechny takové objekty. Níže je uveden výběr z nejpodstatnějších funkcionalit těchto objektů:

- nastavení oprávnění pro CRUD operace a přístupu k jednotlivým polím na základě `Profile` [24] nebo `Permission Set` uživatele [25]
- nastavení viditelnosti jednotlivých záznamů podle role uživatele [26]
- uživatelské rozhraní pro práci se záznamy (stránky pro prohlížení a úpravu)

- zajištění kvality dat pomocí validačních pravidel [27]
- automatizace pomocí `Workflow, Approvals, Flows` nebo `Process Builder` [28]

Vztahy entit jsou zobrazeny na diagramu 2.3.

2.3.1 Subjekt (Individual)

Tato entita reprezentuje subjekt údajů. *Subjekt* je hlavní objekt, ke kterému se přímo či nepřímo váží všechny ostatní entity. U subjektu je evidováno jeho jméno, příjmení, email, telefon a adresa (ulice, PSČ, město, stát). Klient může objekt rozšířit o libovolné další atributy (např. identifikátor).

2.3.2 Žádost (Request)

Entita *Žádost* reprezentuje aplikaci práva nařízení GDPR. U žádosti je evidován typ práva, vazba na `Subjekt`, informace o podání žádosti (datum podání, poznámka, předmět), informace o zpracování (status, zpracovatel, termín, počet úkolů) a odpověď (datum, vyrozumění).

K žádosti je možné ukládat soubory jako přílohu.

2.3.3 Identifikátor subjektu (Individual Identifier)

Entita *Identifikátor subjektu* slouží k evidenci přítomnosti dat subjektu v konkrétním `Informačním Systému`. U identifikátoru je evidována vazba na `Subjekt`, vazba na `Objekt` a konkrétní hodnota identifikátoru.

Na základě záznamu této entity je určeno, kde všude mají být sbírána data subjektu.

2.3.4 Informační subjekt (Information System)

Entita *Informační systém* reprezentuje konkrétní informační systém, jenž uchovává osobní údaje subjektů. U této entity je evidován název, popis, vlastník a příznak, zda je možné data sbírat automaticky.

2.3.5 Objekt (Object)

Entita *Objekt* reprezentuje konkrétní objekt, který obsahuje osobní údaje. Tímto objektem může být například `SObject` v Salesforce nebo tabulka v relační databázi.

U této entity je evidován název, vazba na `Informační systém` a API název, jenž slouží ke zpracování.

Obrázek 2.3: SObject diagram

2.3.6 Osobní údaj (Personal Data)

Entita *Osobní údaj* slouží k popisu umístění konkrétních osobních údajů. Tyto údaje je možné shlukovat do množin pomocí vazby na entitu `Množina osobních údajů`. U každého osobního údaje je evidováno, na základě jakého právního důvodu jsou data zpracovávána.

Dále je u entity evidován název, popis, vazba na `Objekt`, kde jsou uložena data, a druhá vazba na `Objekt`, jež tato data identifikuje (pokud má Subjekt přiřazen identifikátor tohoto objektu, může systém obsahovat jeho osobní údaje).

2.3.7 Množina osobních údajů (Personal Data Set)

Entita *Množina osobních údajů* slouží k shlukování osobních údajů, jež mají společné vlastnosti (např. právní důvod, souhlas apod.). U této entity je evidován pouze název a popis.

2.3.8 Úkol (Task)

Entita *Úkol* reprezentuje zadání úkolu pro vlastníka informačního systému, který má na základě tohoto záznamu provést vyhledání osobních údajů subjektu. U úkolu je evidován typ, jenž určuje, jestli mohou být data sesbírána automaticky. Dále je u této entity evidován zpracovatel, vazba na `Osobní údaj` a konkrétní hodnota identifikátoru, jež je použita pro vyhledání dat.

2.3.9 Data subjektu (Individual's Data)

Entita *Data Subjektu* reprezentuje nalezená data pro `Úkol`. Díky vazbě na Úkol může pro jeden záznam úkolu existovat více vyhledaných dat. U této entity jsou evidovány data a právní důvod zpracování. Data jsou v databázi uložena jako řetězec.

2.3.10 Právní důvod (Lawful Basis)

Entita *Právní důvod* eviduje, na základě jakého právního titulu jsou zpracovávány konkrétní osobní údaje. U této entity je evidován název, popis a příznak určující, zda je pro zpracování osobních dat potřeba zvláštní souhlas subjektu.

2.3.11 Souhlas (Consent)

Entita *Souhlas* reprezentuje souhlas se zpracováním osobních údajů, který vyslovil subjekt. U této entity je evidován pouze název, popis a vazba na `Subjekt`, ale klient si může tuto entitu sám rozšířit o další atributy (např. datum a zdroj souhlasu).

2.3.12 Omezení (Restriction)

Entita *Omezení* slouží k evidenci případů, kdy subjekt vznese *Právo na omezení zpracování*. Toto omezení se vztahuje ke konkrétní množině osobních dat. Dále je u entity evidován název, popis a vazba na `Subjekt`.

2.3.13 Aplikační log (Application Log)

Entita *Aplikační log* slouží k ukládání historie provedených akcí a změn v systému. U této entity jsou ukládány informace, jež popisují kontext, ve kterém byl log vytvořen (zpráva, závažnost, název a metoda třídy, data a případně informace o výjimce).

2.4 Případy užití

V této sekci jsou popsány případy užití (*use cases*) systému, jež definují chování systému z pohledu uživatele a tím popisují funkcionalitu systému.

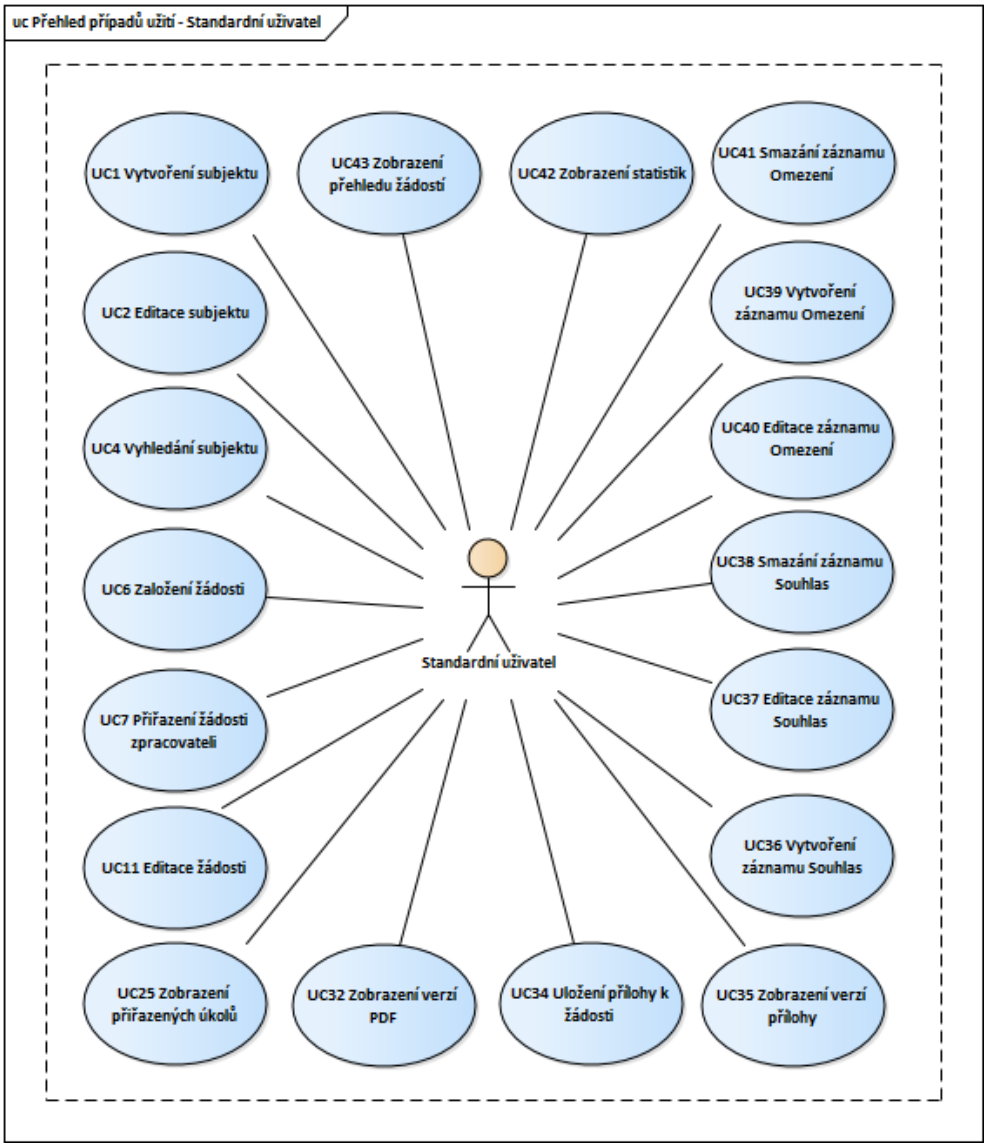
Případy užití spolu s příslušnou rolí uživatele jsou zobrazeny na diagramech 2.4, 2.5 a 2.6. Mapování případů užití na jednotlivé funkční požadavky je zobrazeno v tabulce 2.1.

Tabulka 2.1: Přehled realizace požadavků případy užití

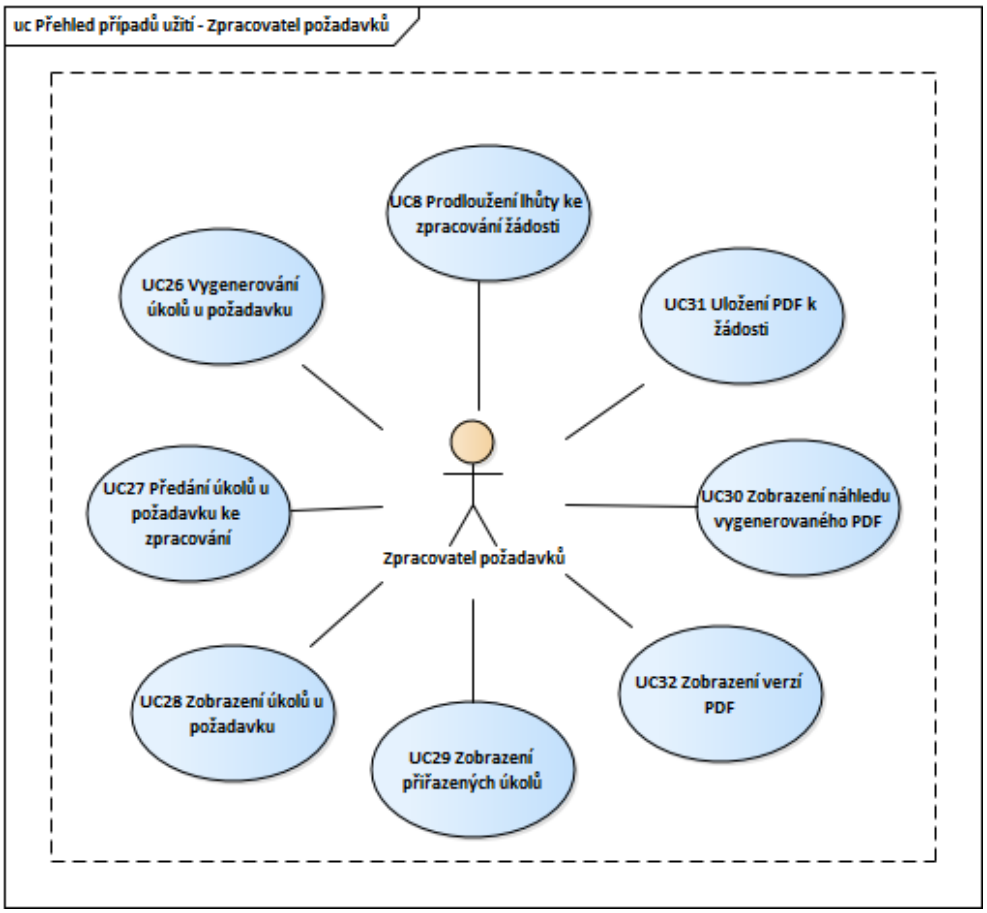
Zkratka	**Název případu užití**	**Požadavek**
UC1	Vytvoření subjektu	F1
UC2	Editace subjektu	F1
UC3	Smazání subjektu	F1
UC4	Vyhledání subjektu	F1
UC5	Nastavení parametrů vyhledávání subjektu	F1
UC6	Založení žádosti	F2
UC7	Přiřazení žádosti zpracovateli	F2
UC8	Prodloužení lhůty ke zpracování žádosti	F2
UC9	Nastavení lhůt pro zpracování	F2
UC10	Nastavení defaultního zpracovatele	F2
UC11	Editace žádosti	F2
UC12	Smazání žádosti	F2
UC13	Vytvoření záznamu Informační systém	F3
UC14	Editace záznamu Informační systém	F3
UC15	Smazání záznamu Informační systém	F3
UC16	Vytvoření záznamu Objekt	F3
UC17	Editace záznamu Objekt	F3
UC18	Smazání záznamu Objekt	F3
UC19	Vytvoření záznamu Osobní údaj	F3

Tabulka 2.1: Přehled realizace požadavků případy užití

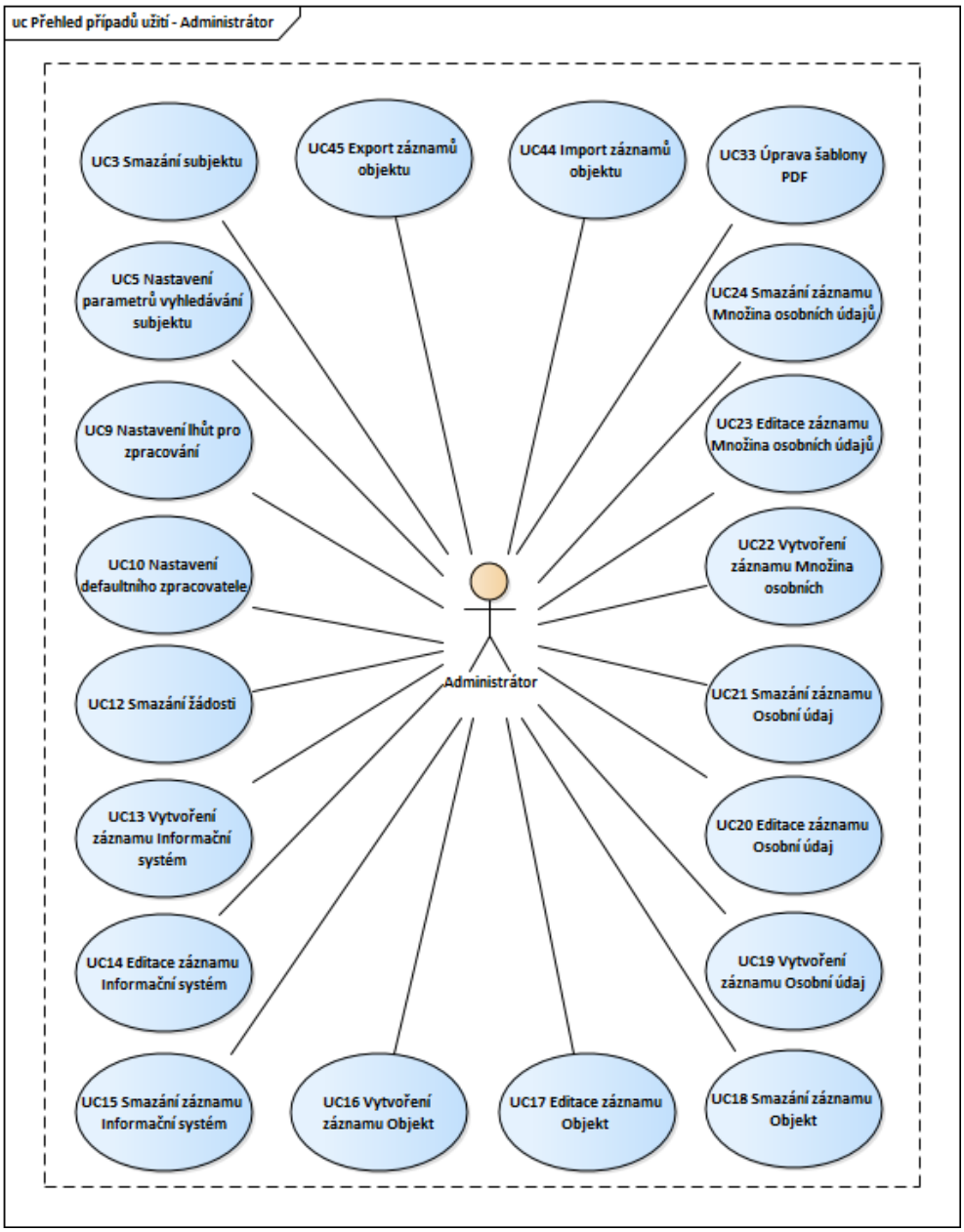
Zkratka	**Název případu užití**	**Požadavek**
UC20	Editace záznamu Osobní údaj	F3
UC21	Smazání záznamu Osobní údaj	F3
UC22	Vytvoření záznamu Množina osobních údajů	F3
UC23	Editace záznamu Množina osobních údajů	F3
UC24	Smazání záznamu Množina osobních údajů	F3
UC25	Zobrazení přiřazených úkolů	F4
UC26	Vygenerování úkolů u požadavku	F4
UC27	Předání úkolů u požadavku ke zpracování	F1
UC28	Zobrazení úkolů u požadavku	F4
UC29	Zobrazení přiřazených úkolů	F4
UC30	Zobrazení náhledu vygenerovaného PDF	F6
UC31	Uložení PDF k žádosti	F6
UC32	Zobrazení verzí PDF	F6
UC33	Úprava šablony PDF	F6
UC34	Uložení přílohy k žádosti	F7
UC35	Zobrazení verzí přílohy	F7
UC36	Vytvoření záznamu Souhlas	F8
UC37	Editace záznamu Souhlas	F8
UC38	Smazání záznamu Souhlas	F8
UC39	Vytvoření záznamu Omezení	F9
UC40	Editace záznamu Omezení	F9
UC41	Smazání záznamu Omezení	F9
UC42	Zobrazení statistik	F10
UC43	Zobrazení přehledu žádostí	F10
UC44	Import záznamů objektu	F11
UC45	Export záznamů objektu	F11



Obrázek 2.4: Přehled případů užití - Standardní uživatel



Obrázek 2.5: Přehled případů užití - Zpracovatel požadavků



Obrázek 2.6: Přehled případů užití - Administrátor

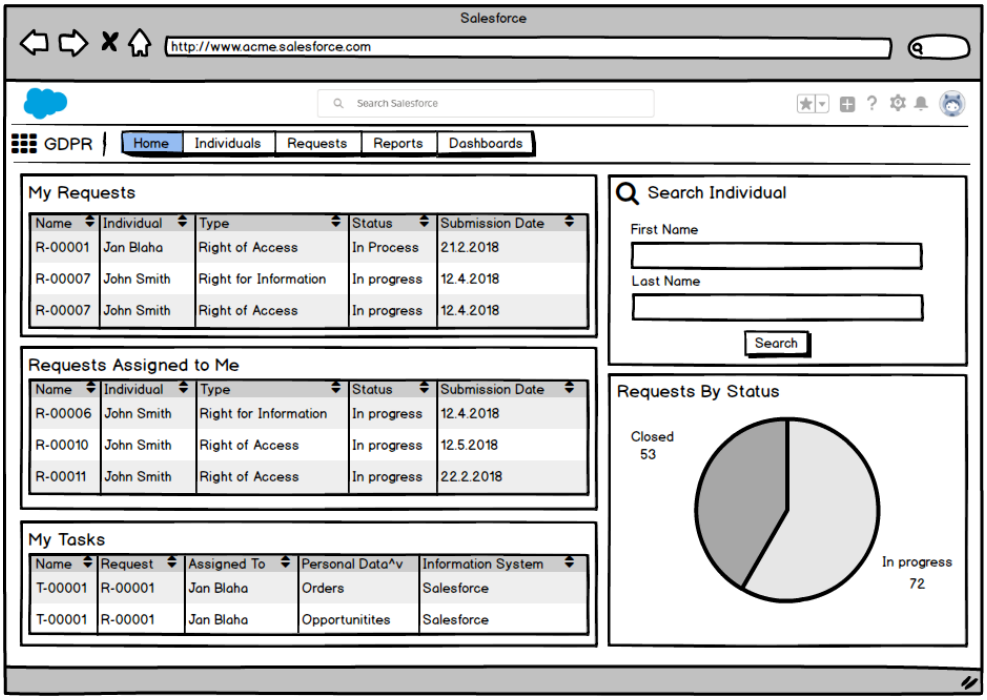
2.5 Návrh uživatelského rozhraní

V této sekci je popsán návrh grafického uživatelského rozhraní (GUI) aplikace. Při návrhu rozhraní byl kladen důraz na zachování standardního vzhledu a ovládání platformy Salesforce. Dále byl kladen důraz na to, aby uživatel měl přístup vždy pouze k údajům, které jsou pro něj v danou chvíli užitečné.

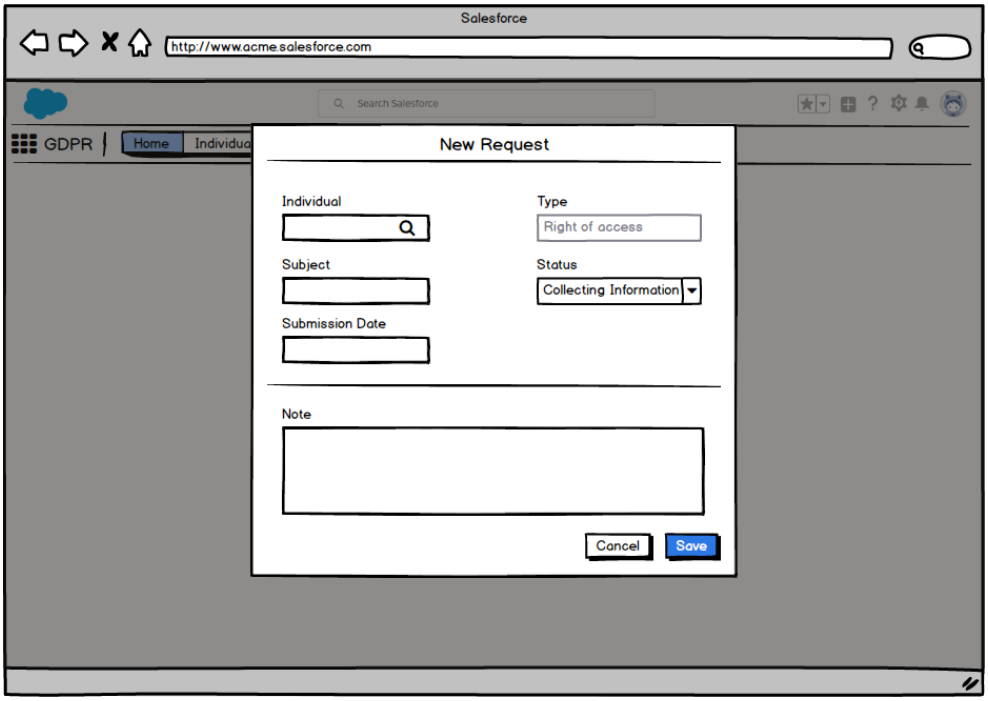
Pro vytvoření náhledu uživatelského rozhraní během návrhu byly vytvořeny wireframy, jejichž cílem je rámcově definovat zobrazované prvky a funkcionality na obrazovce. Pomocí wireframů lze jednoduše odhalit nedostatky rozhraní již ve fázi návrhu. Wireframy lze vytvořit pouze za pomoci tužky a papíru nebo lze využít celé řady specializovaných programů. Pro účely této práce byl použit nástroj Balsamiq [29].

Na obrázku 2.7 je zobrazen wireframe domovské stránky aplikace. Tato stránka je vstupním bodem systému. Na obrazovce je zobrazen přehled žádostí, které podal uživatel sám nebo mu byly přiřazeny (UC42, UC43). Na této stránce je také umístěn formulář pro vyhledávání subjektů (UC4).

Na obrázku 2.8 je wireframe stránky pro založení nové žádosti (UC6). Tato stránka je zobrazena uživateli po zvolení typu žádosti.



Obrázek 2.7: Wireframe: domovská stránka



Obrázek 2.8: Wireframe: založení nové žádosti

2.6 Platforma Salesforce

V této sekci je stručně představena cloudová platforma Salesforce a jsou zde popsány aspekty, které ovlivnily vývoj systému GDPR Podatelna.

Salesforce je největší CRM (Customer Relationship Management) platforma na světě [30], jež nabízí funkcionalitu pro celou řadu odvětví (obchod, marketing, zákaznický servis, interní procesy apod.). Společnost byla založen v roce 1999 s cílem nabízet cloudové CRM pomocí modelu SaaS (*Software as a Service*) [31]. Řešení je vhodné pro společnosti všech velikostí. Pro menší společnosti je navíc v nabídce základní řešení s nižší cenou [32]

Platforma Salesforce se skládá z jednotlivých modulů, z nichž si může klient složit vlastní instanci.

- Sales Cloud – nástroj pro správu obchodních příležitostí
- Service Cloud – nástroj pro řízení zákaznického servisu
- Marketing Cloud – nástroj pro tvorbu individuálního a přesně cíleného obsahu pro potencionální zákazníky
- Community Cloud – nástroj pro sdílení funkcionality systému s dalšími subjekty (např. dodavatelé, zákazníci)

- Analytics – nástroj pro analýzu dat v Salesforce
- IoT Cloud – propojení Salesforce s internetem věcí
- Quip – nástroj pro týmovou spolupráci nad obsahem (dokumenty)
- a další

Salesforce je možné používat přes libovolný webový prohlížeč. Pro použití na mobilním telefonu nebo tabletu lze využít aplikaci *Salesforce* pro operační systémy iOS a Android [33].

2.6.1 Licence

Platforma je zpoplatněna po jednotlivých modulech za každého aktivního uživatele, který má přístup do systému. Průměrná cena takové licence je přibližně 150 USD za měsíc. Detailní informace k ceně a funkcionalitě jednotlivých modulů jsou dostupné na stránkách společnosti [34].

Vzhledem k tomu, že aplikace GDPR Podatelna obsahuje Apex kód, který bude klient ve své instanci rozšiřovat, je pro její instalaci a použití potřeba vlastnit licenci *Lightning Enterprise* nebo vyšší.

2.6.2 Salesforce a GDPR

Pro společnost Salesforce je ochrana dat klientů jednou z největších priorit. Po schválení nařízení GDPR byl založen na webových stránkách portál poskytující informace potřebné pro naplnění souladu s nařízením [35]. Použití platformy Salesforce je samo o sobě v souladu s nařízením, ale na klientech zůstává povinnost zajistit legitimní zacházení s osobními údaji v jeho instanci.

Salesforce v poslední aktualizaci (Spring '18) vydal nástroj pro zpracování základních informací o subjektech osobních údajů [36]. V této aktualizaci byl přidán nový objekt `Individual`, který je navázán na objekty typu `Lead`, `Contact` a `Person Account`. Pomocí tohoto objektu lze ukládat základní souhlasy subjektu se zpracováním. Dále je možné evidovat, jaké záznamy třech výše uvedených objektů patří k danému subjektu. Tato funkcionalita může být zapnuta klientem v nastavení své instance.

Aplikace GDPR Podatelna nebude využívat standardní objekt `Individual`, jelikož se nejedná o plnohodnotný objekt. Není pro něj možné například definovat Record Type, přidat tlačítka a akce, upravovat stránky záznamů v Lightningu nebo napsat Trigger pomocí Apexu (popis viz kapitola 3.1).

2.6.3 Distribuce aplikace

Na platformě je možné distribuovat funkcionalitu do jednotlivých instancí pomocí *package*. Package je kontejner, který je možný využít pro distribuci řešení, které se skládá z jednotlivých komponent platformy. Využít lze jak pro

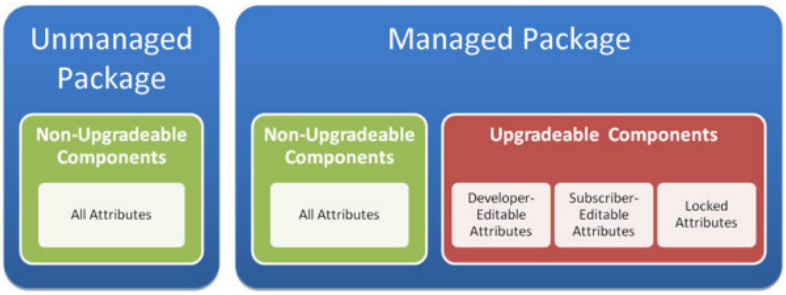
malé aplikace obsahující několik reportů, tak i pro velká řešení skládající se z několika aplikací.

Package může být vytvořen z jakékoliv *Developer Edition* nebo *Partner Developer Edition* instance [37]. Každý vývojář si může zdarma zřídit libovolný počet vlastních *Developer Edition* instancí. Tato instance má řadu omezení (např. počet uživatelů) a je určena pro studijní účely nebo vývoj jednoduchých aplikací. *Partner Developer Edition* instanci mohou vytvořit partnerské společnosti, jež spolupracují se Salesforce. Instance tohoto typu byla použita pro vývoj systému GDPR Podatelna.

Pro distribuci funkcionality lze zvolit z dvou typů package - *managed* nebo *unmanaged*. Každý typ je vhodné použít v jiném případě a je mezi nimi řada rozdílů. Managed i unmanaged package lze nahrát na obchod aplikací AppExchange [20].

2.6.3.1 Unmanaged package

Unmanaged package je typicky vhodné použít pro distribuci open-source projektů nebo aplikací, u kterých je žádoucí, aby v cílové instanci bylo možné upravovat jednotlivé komponenty (např. úprava Apex tříd nebo Visualforce stránek). Jakmile je package jednou nainstalován, nemá vývojář možnost jakkoliv změnit funkcionalitu systému (např. pomocí aktualizací). Dále je potřeba zohlednit, že vývojář nemá možnost skrýt detaily implementace před zákazníkem. Z toho důvodu není vhodné použít unmanaged package pro aplikace, kde jsou detaily implementace obchodním tajemstvím vývojářské společnosti.



Obrázek 2.9: Srovnání managed a unmanaged package

2.6.3.2 Managed package

Managed package je vhodné použít pro aplikace, které vývojář průběžně spravuje. Aplikaci lze průběžně rozvíjet nebo upravovat, a klient má po nainstalování možnost provádět aktualizaci produktu.

Pokud je package publikován v obchodě AppExchange, může vývojář využít finanční model společnosti Salesforce, kterým lze jednoduše zpoplatnit aplikaci.

Aplikace bývají zpoplatněny měsíčním poplatkem za každého uživatele, kterému je aplikace zpřístupněna.

Při vytváření managed package je potřeba nastavit unikátní prefix, který se použije pro API názvy všech komponent a tím je zajištěno, že při instalaci nemůže dojít ke konfliktu s jinými aplikacemi. Na rozdíl od unmanaged package může vývojář odstínit klienty od detailů implementace a tím chránit své obchodní tajemství.

Některé komponenty managed package lze průběžně aktualizovat, jiné nikoliv. Detailní popis jednotlivých komponent je dostupný v dokumentaci pro partnery Salesforce [38]. Tato omezení je potřeba zohlednit v průběhu implementace.

Systém GDPR Podatelna bude distribuován jako managed package, jelikož bude v budoucnu dále rozvíjen a upravován. Zároveň není žádoucí, aby měli klienti přístup k detailům implementace a mohli upravovat libovolné komponenty.

2.6.4 Limity

Jednotlivé klientské instance jsou provozovány na serverech společnosti Salesforce, které využívají *Multitenant* architekturu [39]. V této architektuře je umístěno na jednom fyzickém serveru více klientských instancí a dochází ke sdílení výpočetních prostředků mezi instancemi.

Aby mohl Salesforce zajistit, že instance klientů budou fungovat konzistentním a správným způsobem i s doprogramovanými rozšířeními, je vývoj omezen sadou regulací a limitů. Limity zajišťují, aby vývojáři během vývoje dodržovali *best practices* jednotlivých nástrojů, a aby nedocházelo k degradaci výkonu serveru.

V následující části jsou shrnuty nejzásadnější limity, které je potřeba zohlednit v průběhu implementace. Podrobný popis všech limitů a omezení je dostupný v dokumentaci Salesforce [40].

- 100 SOQL dotazů do databáze během transakce (pro asynchronní transakce je limit dvojnásobný)
- 50 000 záznamů ve výsledcích SOQL dotazů během transakce
- 2 000 záznamů ve výsledku jednoho SOQL dotazu
- 150 DML operací během transakce
- 10 000 záznamů v jedné DML operaci
- 100 volání externích webových služeb v jedné transakci
- 120 vteřin pro volání externích služeb celkem v jedné transakci

- 6 MB pro požadavek nebo odpověď při volání externí služby (pro asynchronní transakce je limit dvojnásobný)
- 50 naplánovaných úloh (`System.enqueueJob`) v jedné transakci
- 10 odeslaných emailů (metoda `sendEmail`) v jedné transakci
- 6 MB pro alokování paměti transakce (pro asynchronní transakce je limit dvojnásobný)
- délka transakce maximálně 10 s (pro asynchronní je limit 60 s)
- 250 000 volání serveru z prostředí Lightning za 24 hodin
- 10 současně běžících transakcí, které byly spuštěny z Lightningu a běží déle než 5 vteřin
- 120 s pro provedení SOQL dotazu
- 15 MB pro velikost Visualforce stránky
- 10 MB pro velikost nahraného souboru přes Visualforce stránku
- 135 KB pro velikost *view state* Visualforce stránky
- 50 000 získaných záznamů z databáze v dotazu z Visualforce stránky
- 1 000 záznamů v kolekci, přes kterou je iterováno na Visualforce stránce (10 000 pouze pro čtení)

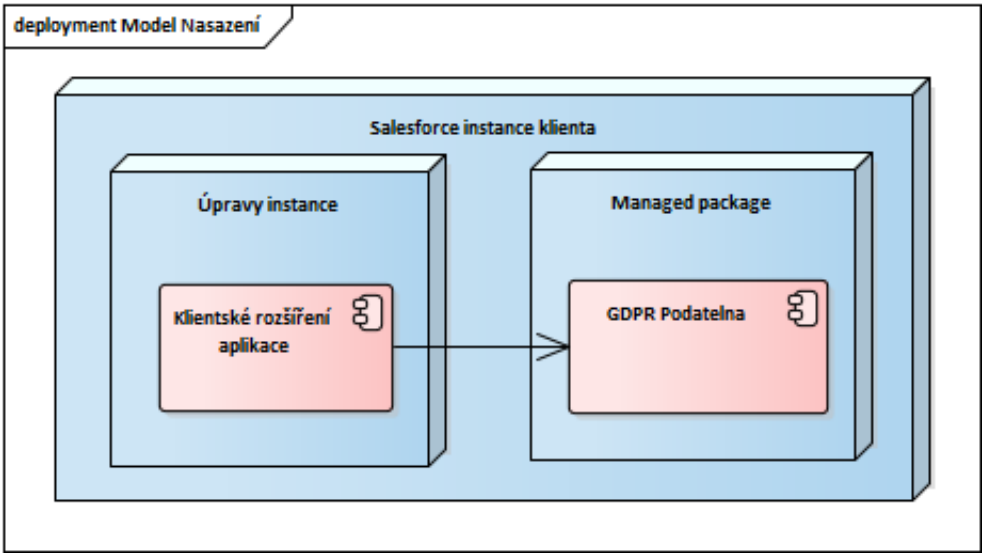
2.7 Nasazení aplikace

Systém GDPR Podatelna bude do jednotlivých Salesforce instancí klientů distribuován jako *managed package* (popis viz 2.6.3.2).

Instalace aplikace může být provedena pomocí vygenerovaného odkazu z vývojářské instance. Po případném publikování aplikace v obchodě AppExchange, bude možné aplikaci nainstalovat bez interakce s vývojářem. Při instalaci klient zvolí, pro jaké uživatele bude aplikace přístupná. Vzhledem k tomu, že v cílové instanci má klient své vlastní nastavení profilů uživatelů, musí při instalaci provést mapování rolí uživatelů aplikace GDPR Podatelna na jednotlivé profily.

Funkcionalita aplikace může být klientem dále rozvíjena a upravována pro jeho potřeby. Klient bude mít možnost některé části aplikace upravit změnou implementací rozhraní, jež budou veřejně dostupné (např. způsob generování PDF, získávání osobních údajů z externích systémů apod.). Dále bude mít klient možnost pro jednotlivé entity přidávat ukládané atributy nebo měnit jejich obrazovky (např. úprava procesu zpracování jednoho typu žádostí subjektu).

Na obrázku 2.10 je zobrazen model nasazení aplikace.



Obrázek 2.10: Model nasazení aplikace

2.8 Předpoklady pro systém

V této sekci jsou popsány podmínky, které musí být splněny, aby systém GDPR Podatelna pracoval korektně a bylo tak dosaženo souladu s nařízením GDPR.

Systém GDPR Podatelna je nástroj pro podporu procesu sbírání a zpracování žádostí od subjektů osobních údajů. Systém se nezabývá správou a údržbou osobních údajů.

Pro správné fungování systému a zpracování jednotlivých žádostí je potřeba, aby jednotlivé informační systémy, jež zpracovávají osobní údaje, splňovaly následující kritéria:

- **Systém ukládá pouze minimální množinu osobních údajů**

  Každý informační systém ukládá a zpracovává pouze takové osobní údaje, které jsou nezbytné pro jeho chod.

- **Systém ukládá pouze relevantní osoby**

  Každý informační systém ukládá pouze údaje osob, které jsou pro daný systém relevantní (např. HR systém ukládá pouze údaje zaměstnanců a žádných jiných subjektů).

- **Systém udržuje aktuální údaje**

  Pokud informační systém získává data z jiných systémů (např. synchronizací) a není primárním zdrojem dat, musí zajistit, aby jeho kopie dat byla aktuální.

- **Systém používá údaje oprávněně**

  Každý informační systém musí zajistit, že zpracování osobních údajů probíhá v souladu s jejich právním důvodem.

- **Systém maže nepotřebné osobní údaje**

  V každém informačním systému musí dojít ke smazání konkrétních osobních údajů v případě, že pomine poslední právní důvod jejich zpracování.

- **Systém poskytuje data aplikaci GDPR Podatelna**

  Všechny informační systémy musí poskytnout přístup ke svým datům aplikaci GDPR Podatelna. Přístup může být manuální nebo automatizovaný pomocí API.

- **Systém je schopen pozastavit zpracování**

  Systém musí respektovat případná omezení zpracování konkrétních osobních údajů a případně dočasně pozastavit jejich zpracování.

- **Testování a vývoj neprobíhá na produkčních datech**

  Při vývoji nebo testování nové funkcionality informačního systému by nemělo docházet k použití produkčních dat, jež obsahují osobní údaje klientů, pokud k tomu nemá společnost schválenou výjimku.

Kapitola 3

Realizace

Tato kapitola se zabývá implementací systému GDPR Podatelna, jenž byl vyvinut na základě návrhu řešení v předchozí části. Součástí kapitoly je popis použitých technologií, které byly zvoleny na základě analýzy požadavků. V další části kapitoly jsou rozebrány detaily implementace stěžejní funkcionality systému. V závěru kapitoly je přiložena ukázka uživatelského rozhraní vytvořeného systému.

3.1 Technologie

Základní sada použitých nástrojů byla dána již samotným návrhem systému. Vzhledem k tomu, že systém byl vyvíjen jako managed package na platformě Salesforce, byly využity především nástroje, jež tato platforma nabízí.

3.1.1 Salesforce

Platforma Salesforce nabízí celou řadu vývojářských nástrojů, kterými je možné jednotlivé instance klientů upravovat na míru. Vzhledem k tomu, že platforma byla postavena na technologiích ekosystému Java, jsou Salesforce technologie částečně podobné svým ekvivalentům v Javě.

3.1.1.1 Apex

Apex je objektově orientovaný programovací jazyk, pomocí kterého lze vyvíjet aplikace a rozšíření na platformě Salesforce. Kód jednotlivých tříd je po odeslání do vývojářské instance přeložen do Java *bytecode* [41] a následně je interpretován virtuálním strojem serveru. Z toho důvodu je syntaxe jazyka Apex velmi podobná syntaxi jazyku Java. Srovnání těchto dvou jazyků je dostupné v dokumentaci jazyka Apex [42]. Kód 3.1 obsahuje ukázku třídy napsané v Apexu.

Kód 3.1: Ukázka třídy v jazyce Apex (*AssignRequestController.cls*)

```
public class AssignRequestController {

   final private Id requestId;
   public String errorMessage { get; set; }

   public AssignRequestController(ApexPages.StandardController
       standardCtrl) {
      errorMessage = '';
      requestId = standardCtrl.getId();
   }

   public PageReference assignRequest() {
      try {
         ...
         // method code here
         ...
      } catch(Exception ex) {
         errorMessage = Label.Error_Saving_Data + ': ' +
             ex.getMessage() + '\n';
      }
      return new PageReference('/' + requestId);
   }

}
```

3.1.1.2 Visualforce

Visualforce je framework pro tvorbu grafického uživatelského rozhraní na platformě Salesforce [43]. Jedná se o značkovací jazyk, který slouží k definování způsobu renderování HTML dokumentů. Samotné renderování jednotlivých stránek probíhá na straně serveru.

Každá stránka může mít definovaný svůj vlastní *controller*. Controller slouží k tvorbě dynamického obsahu stránky a poskytuje funkcionalitu pro interakci uživatele se stránkou. Salesforce poskytuje pro každý SObject jeho vlastní standardní controller se základní funkcionalitou pro práci s daným záznamem (zobrazení atributů, uložení záznamu do databáze apod.). Vývojář má možnost pro své potřeby vytvořit vlastní controller, který může rozšiřovat ten standardní.

Tento framework byl při vývoji systému použit pro implementaci stránek, jež obsahují složitější funkcionalitu systému (např. generování PDF dokumentů).

Kód 3.2 obsahuje ukázku stránky, která slouží k přiřazení žádosti zpracovateli. Tato stránka je přístupná ze záznamu žádosti přes *Detail Page Button* [44]. Stránka po svém načtení ihned provede Apex kód a po uložení dat je

uživatel přesměrován zpět na stránku záznamu žádosti.

Kód 3.2: Ukázka Visualforce stránky (*AssignRequest.page*)

```
<apex:page standardController="Request__c"
    extensions="AssignRequestController" action="{!assignRequest}"
    sidebar="false">

  <apex:stylesheet value="{!URLFOR($Resource.slds_252,
    '/assets/styles/salesforce-lightning-design-system-vf.css')}"/>

  <script>
    // javascript code
  </script>

  <style>
    // css styles
  </style>

  <apex:form styleClass="slds container">
    <c:Alert />
    <apex:outputPanel styleClass="hidden errorMessage">
      {!errorMessage}
    </apex:outputPanel>
    <apex:commandButton
      styleClass="backButton slds-button slds-button_neutral"
      value="{!$Label.Back}" action="{!cancel}"/>
  </apex:form>
</apex:page>
```

3.1.1.3 Lightning Component Framework

Lightning je nové uživatelské prostředí platformy Salesforce, na které postupně přechází většina instancí. Lightning Component Framework je nástroj pro tvorbu aplikací a komponent pro uživatelské prostředí Lightning [45]. Pomocí tohoto frameworku lze vytvářet responzivní aplikace, které lze spustit ve webovém prohlížeči, na mobilních telefonech a tabletech.

Na rozdíl od Visualforce dochází při použití tohoto frameworku k přesunu logiky aplikací na stranu klienta a tím je snížena zátěž serverů. Lightning je postaven na programovacím jazyku JavaScript. Nástroj Lightning se skládá ze dvou základních celků - *Components* a *Events*.

Components jsou nezávislé a znovupoužitelné prvky uživatelského prostředí (např. formulář, tabulka s výsledky apod.). Salesforce nabízí celou škálu základních komponent (např. prvky formulářů, tlačítka, nabídky, tabulka s daty), popřípadě nabízí prostor k implementaci vlastních. Vytvořená komponenta

je definována pomocí speciálního značkovacího jazyka ve formátu XML, jenž definuje strukturu komponenty. Každá komponenta může mít 2 controllery - jeden na straně klienta v programovacím jazyce Javascript a druhý na straně serveru naimplementovaný v jazyce Apex. Dále může být součástí komponenty soubor s CSS styly a soubor s dokumentací.

Events (události) slouží k lepšímu oddělení vazeb mezi komponenty. Jednotlivé komponenty mohou být mezi sebou notifikovány o provedených změnách. Events je vhodné použít například v situaci, pokud dojde ke změně dat záznamu a je potřeba upozornit jiné komponenty, aby provedly určitou akci (např. překreslení dat).

Při implementaci systému GDPR Podatelna byl využit Lightning na vývoj formuláře pro vyhledávání subjektů.

Kód 3.3 obsahuje ukázku jednoduché komponenty, která slouží k zobrazení seznamu záznamů.

Kód 3.3: Ukázka Lightning komponenty

```
<aura:component controller="ContactListController">

   <aura:attribute name="subjects" type="Subject__c[]"/>
   <aura:handler name="init" value="{!this}" action="{!c.doInit}"
       />

   <ul>
      <aura:iteration items="{!v.subjects}" var="subject">
         <li>
            <a href="{! '#/sObject/' + subject.Id + '/view'}">
               <p>{!subject.First_Name__c}</p>
               <p>{!subject.Last_Name__c}</p>
            </a>
         </li>
      </aura:iteration>
   </ul>

</aura:component>
```

3.1.1.4 Lightning Design System

Lightning Design System (SLDS) je oficiální CSS framework společnosti Salesforce [46]. Framework umožňuje tvorbu webových aplikací, jejichž uživatelské rozhraní vypadá a používá stejné prvky jako uživatelské rozhraní Lightning prostředí.

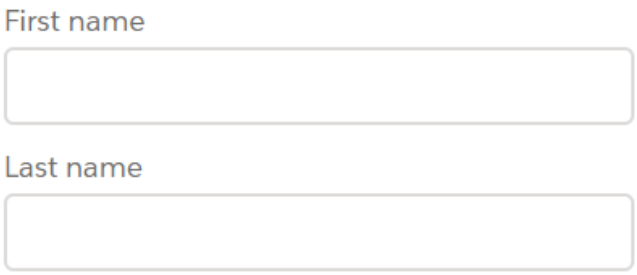
Framework se skládá z definic CSS tříd, jež se přidávají na stylované prvky. Na webových stránkách frameworku je ukázka komponent, které je možné

využít [47].

Kód 3.4 obsahuje ukázku využití SLDS frameworku. Obrázek 3.1 znázorňuje grafickou podobu formuláře definovaného pomocí kódu 3.4

Kód 3.4: Ukázka SLDS frameworku

```
<div class="slds-form slds-form_stacked">
  <div class="slds-form-element">
    <label class="slds-form-element__label"
        for="input-first-name">First name</label>
    <div class="slds-form-element__control">
      <input type="text" id="input-first-name"
          class="slds-input"/>
    </div>
  </div>
  <div class="slds-form-element">
    <label class="slds-form-element__label"
        for="input-last-name">Last name</label>
    <div class="slds-form-element__control">
      <input type="text" id="input-last-name"
          class="slds-input"/>
    </div>
  </div>
</div>
```



Obrázek 3.1: Ukázka SLDS frameworku - náhled

3.1.2 jQuery

jQuery je javascriptová knihovna pro usnadnění interakce mezi HTML a jazykem JavaScript [48]. Knihovna je volně šiřitelný software a je vydána pod licencí MIT.

Při implementaci byla tato knihovna využita pro usnadnění manipulace s DOM (*Document Object Model*) stromem stránek naimplementovaných pomocí Visualforce.

3.1.3 pdfmake

Pdfmake je open-source knihovna pro generování PDF dokumentů v jazyce JavaScript [49]. Definice dokumentu se provádí deklarativním způsobem. Framework nabízí celou škálu definovatelných komponent (např. tabulky, seznamy, odstavce, nadpisy, obrázky), které lze upravovat pomocí stylů. Vygenerovaný dokument je možné otevřít v novém okně, vytisknout nebo stáhnout.

Knihovna je zdarma a volně k použití pod licencí MIT [50].

Při implementaci systému byla tato knihovna použita pro generování PDF odpovědí na žádosti subjektů.

Kód 3.5 obsahuje ukázku generování jednoduchého PDF pomocí této knihovny.

Kód 3.5: Ukázka generování PDF pomocí pdfmake

```
<html>
  <head>
    <meta charset='utf-8'>
    <script src='pdfmake.min.js'></script>
    <script src='vfs_fonts.js'></script>
  </head>
  <body>
  <script>
    var docDefinition = { content: 'This is an sample PDF printed
        with pdfMake' };

    pdfMake.createPdf(docDefinition).open();

    pdfMake.createPdf(docDefinition).print();

    pdfMake.createPdf(docDefinition).download('fileName.pdf');
  </script>
  </body>
</html>
```

3.2 Implementace

V této sekci jsou popsány stěžejní části implementace systému GDPR Podatelna. Pro popis byly vybrány části, při kterých bylo využito netypických postupů nebo technologií.

3.2.1 Návrh tříd

Platforma Salesforce bohužel neumožňuje rozdělit třídy do jednotlivých balíčků (*packages*). Z toho důvodu nelze vytvořit strukturu Apex tříd. Při návrhu byl kladen důraz především na dodržení *best practices* platformy. Všechny naimplementované třídy a jejich metody jsou řádně zdokumentovány.

V kapitole 2.1je u několika funkčních požadavků (F1, F2, F4, F5, F6) uvedeno, že klient, jenž si systém nainstaluje, bude mít možnost rozšířit nebo dodefinovat funkcionalitu systému. Některá tato rozšíření budou implementována pomocí jazyku Apex. Systém tedy bylo potřeba navrhnout tak, aby tyto změny byly možné.

Dependency injection

Při implementaci systému nebyla známa konkrétní rozšíření klientů, tudíž na nich nemůže být systém závislý. Zároveň u *managed package* není možné upravovat Apex třídy v klientských instancích.

Z tohoto důvodu bylo potřeba zajistit mechanismus *dependency injection* [51]. Pomocí tohoto mechanismu lze odstínit konzumenta rozhraní (dependency) od přímého získání reference na konkrétní implementaci (injection).

Platforma Salesforce nenabízí žádný způsob jak dependency injection zajistit, a tak bylo potřeba naimplementovat vlastní mechanismus správy závislostí.

Součástí jazyka Apex je třída `Type`, jejíž instance může být vytvořena dynamicky podle jména pro každou Apex třídu [52]. Tato třída má metodu *newInstance()*, která vytvoří novou instanci. Pomocí této funkcionality byl realizován mechanismus dependency injection.

Každá třída, jež bude instanciována tímto mechanismem, musí mít bezparametrický konstruktor, jelikož pomocí výše uvedené metody nelze vytvořit instanci konstruktorem s parametry.

Pro definici vazeb mezi rozhraními a třídami byly vytvořeny *Custom Settings* `Class to Interface Assignment`, kde je pro každé rozhraní možné definovat název třídy, jejíž instance má být vytvořena. Pokud není jméno třídy nastaveno, použije se defaultní implementace rozhraní.

Logika pro získávání závislostí je obsažena ve třídě `ClassUtils`. Tato třída má jednu veřejnou metodu *getImplementation(String interfaceName)*. V metodě je pro název rozhraní z Custom Settings načten název třídy, jejíž instance má být vytvořena. Výsledkem metody je instance implementace přetypovaná na typ `Object`. Konzument musí dle kontextu provést přetypování na konkrétní typ rozhraní.

Kód 3.6 obsahuje proces načtení názvu třídy implementace a vytvoření její instance.

Kód 3.6: Princip načtení závislosti (*ClassUtils.cls*)

```
public static Object getImplementation(String interfaceName) {
   // getting class name
   String className =
       loadClassNameFromCustomSettings(interfaceName);
   if (className == null) {
      className = getDefaultClassName(interfaceName);
   }

   // trying to create instance of classs
   try {
      Object o~= Type.forName(className).newInstance();
      return o;
   } catch(TypeException ex) {
      throw new IllegalArgumentException('Defined class <' +
          className + '> for interface <' + interfaceName + '> is
          not valid.', ex);
   }
}
```

Kód 3.7 obsahuje ukázku použití třídy *ClassUtils* pro získání závislosti konzumentem.

Kód 3.7: Ukázka získání závislosti

```
...
TaskService ts = (TaskService)
    ClassUtils.getImplementation(TaskService.class.getName());
...
```

3.2.2 Zpracování šablon

Součástí funkčního požadavku *F6 – Generování dokumentů* je možnost definovat vzhled generovaných PDF dokumentů pomocí šablon (viz sekce 2.1). Salesforce nabízí možnost definovat šablony pro odesílání emailů (*Email Templates*) [53]. Tato funkcionalita byla využita k definici Javascript kódu, pomocí kterého je PDF dokument vygenerován.

Tyto šablony obsahují definici struktury a vzhledu dokumentu. Jelikož bylo potřeba zajistit, aby jednotlivé dokumenty obsahovaly dynamický obsah pro konkrétní žádosti, byla naimplementována třída *TemplateParser*, jež obsahuje funkcionalitu pro doplnění konkrétních hodnot do šablony. Pro definici umístění jednotlivých hodnot je potřeba přidat do šablony značky (např. *{!request.Type}*). V konstruktoru třídy je předáno jméno šablony pro načtení z databáze. Pro samotné doplnění hodnot třída poskytuje metodu *mergeEle-*

ment s parametry *String elementName* a *Object data*. Tato metoda provede nahrazení všech výskytů značek v šabloně hodnotou dat. Způsob vykreslení dat je určen implementací `RenderElementStrategy` rozhraní. Pro vykreslení dat jsou poskytnuty tři základní implementace - vykreslení řetězce, SObjectu a kolekce. Způsob vykreslení dat může vývojář definovat předáním implementace rozhraní metodě *mergeElement*. Pokud není implementace specifikována, určí se defaultní implementace na základě typu dat. Po vykreslení všech elementů je možné získat hodnotu vyplněné šablony pomocí metody *getBody()*.

Kód 3.8 obsahuje ukázku implementace rozhraní `RenderElementStrategy`, konkrétně třídu `ListRender` pro vykreslení kolekce. Parametrem konstruktoru je oddělovač jednotlivých prvků a strategie pro jejich vykreslení.

Kód 3.8: Ukázka parsování šablon (*ListRender.cls*)

```
public class ListRender implements RenderElementStrategy {
   private final String separator;
   private final RenderElementStrategy listElementStrategy;

   public ListRender(RenderElementStrategy listElementStrategy,
       String separator) {
      this.separator = separator;
      this.listElementStrategy = listElementStrategy;
   }
   public String render(String elementName, Object data, String
       body) {
      checkElementNameNotEmpty(elementName);
      if ((data instanceof List<Object>) == false) {
         throw new IllegalArgumentException('Passed data is not of
             List type.');
      }

      String value = '';
      if (data != null) {
         List<Object> dataList = (List<Object>) data;

         for (Integer i = 0; i<dataList.size(); i++) {
            value += listElementStrategy.render('tmp', dataList[i],
                '{!tmp}');
            if (i<dataList.size() - 1) {
               value += separator;
            }
         }
      }
      return body.replace('{!' + elementName + '}', value);
   }
}
```

Kód 3.9 obsahuje ukázku použití třídy `TemplateParser`. Pomocí tohoto kódu jsou získány definice PDF dokumentů na základě typy žádosti.

Kód 3.9: Parsování definice PDF dokumentu

```
...
TemplateParser parser = new
    TemplateParser(DEFAULT_RESPONSE_TEMPLATE_NAME);
parser.mergeElement('individualFirstName',
    request.Individual__r.First_Name__c);
parser.mergeElement('individualLastName',
    request.Individual__r.Last_Name__c);
parser.mergeElement('companyLogoBase64', companyLogoBase64);

parser.mergeElement('submissionDate', submissionDate);
parser.mergeElement('responseDate', responseDate);
parser.mergeElement('answer', answer);

parser.mergeElement('requestType', request.RecordType.Name);
parser.mergeElement('createdByName', request.CreatedBy.Name);

String pdfDefinition = parser.getBody();
...
```

Načítání šablon

Načítání šablon z databáze pro potřeby parsování definic může probíhat velmi často (např. pro každý řádek tabulky). Aby nedocházelo ke zbytečnému zatěžování databáze a zároveň nedošlo k překročení limitu platformy (100 SOQL dotazů během jedné transakce - viz kapitola 2.6.4), je potřeba si dočasně ukládat v paměti jednotlivé šablony. Tato funkcionalita je poskytována jednoduchou třídou `EmailTemplateLoader`, jež si jednotlivé šablony při prvotním načtení uloží do kolekce pro pozdější použití. Kód 3.10 obsahuje definici třídy `EmailTemplateLoader`.

3.2.3 Generování PDF

Pro zpřístupnění funkcionality generování PDF dokumentů byla naimplementována Visualforce stránka `GenerateRequestPdf`. Tato stránka slouží k zobrazení náhledu vygenerovaného PDF a jeho případného uložení k záznamu žádosti.

Generování PDF je provedeno na straně klienta v prohlížeči pomocí javascriptové knihovny pdfmake (popis viz 3.1.3).

Pro stránku `GenerateRequestPdf` byl naimplementován Apex controller `GenerateRequestPdfController`, jenž rozšiřuje standardní controller objektu

Kód 3.10: Načítání šablon (*EmailTemplateLoader.cls*)

```
public class EmailTemplateLoader {

   private static Map<String, EmailTemplate> devNameToTemplateMap
       = new Map<String, EmailTemplate> ();

   public static EmailTemplate getEmailTemplate(String
       developerName) {
      if (!devNameToTemplateMap.containsKey(developerName)) {
         List<EmailTemplate> emailTemplates = [
            SELECT Id, Body, HtmlValue, Subject
            FROM EmailTemplate
            WHERE DeveloperName = :developerName];
         if (emailTemplates.isEmpty()) {
            throw new IllegalArgumentException('Email template for
                passed developer name (' + developerName + ') does
                not exist.');
         }
         devNameToTemplateMap.put(developerName, emailTemplates[0]);
      }

      return devNameToTemplateMap.get(developerName);
   }
}
```

`Request__c`. V tomto controlleru je pro získání definice PDF použito rozhraní `GeneratePdfDefinitionService`. Toto rozhraní slouží k získání definice PDF pro konkrétní typ žádosti. Defaultní implementace rozhraní může být v případě potřeby klienta nahrazena jinou implementací.

Zpracovaná šablona PDF dokumentu musí obsahovat deklaraci javascriptové proměnné *docDefinition*. Způsob zpracování definice PDF v Javascriptu je znázorněn v kódu 3.11.

Kód 3.11: Zpracování definice PDF (*GenerateRequestPdf.page*)

```
try {
   eval('{!JSENCODE(documentDefinitionJavascript)}');
} catch(err) {
   console.log('error during parsing code for docDefinition:',
       err);
}
if(typeof docDefinition === 'undefined'){
   console.log('ERROR: docDefinition is not defined.');
}
```

Kód 3.12 obsahuje ukázku zpracování PDF. Jsou zde definovány dvě metody. První slouží k vytvoření náhledu PDF. Druhá metoda vytvoří PDF a zavolá Apex controller, kde dojde k uložení dokumentu.

Kód 3.12: Vytvoření PDF (*GenerateRequestPdf.page*)

```
function generatePdfView() {
   pdfMake.createPdf(documentDefinitionGlobal).getDataUrl(function(
       outDoc) {
      if(typeof outDoc !== 'undefined') {
         document.getElementById('pdfViewIframe').src = outDoc;
      }
   });
}

function beforeSavePdf(){
   startLoading('{!JSENCODE($Label.Saving_pdf_Lbl)} ...');
   pdfMake.createPdf(documentDefinitionGlobal).getBase64(function(
       encodedString){
      savePdf(encodedString);
   });
}
```

3.2.4 Automatické sbírání dat subjektu

Pro některé typy žádostí (*Právo na přístup* a *Právo na přenositelnost*) je potřeba provádět sběr zpracovávaných osobních údajů subjektu.

Na základě metadat v záznamech objektu `Personal_Data__c` jsou vygenerovány úkoly (`Task__c`), pomocí kterých správci jednotlivých informačních systémů provádí sběr dat subjektu. Generování úkolů pro konkrétní žádost je přístupné přes Visualforce stránku `CreateTasks`. Tato stránka využívá rozhraní `TaskService`, jež poskytuje funkcionalitu pro generování úkolů. Defaultní implementace vygeneruje pro všechny `Personal_Data__c` záznamy odpovídající úkoly v případě, pokud je pro subjekt, jež podává žádost, definován identifikátor pro příslušný systém. Takto vygenerované úkoly mají nastavený typ podle toho, jakým způsobem mohou být dále zpracovávány - automaticky, nebo manuálně.

Pro předání automatických úkolů ke zpracování slouží Visualforce stránka `ProcessTasks`. Tato stránka naplánuje zpracování úkolů. V případě neúspěchu zobrazí uživateli chybovou hlášku.

Vzhledem k tomu, že k jedné žádostí může být vygenerováno několik desítek až stovek úkolů, nelze úkoly zpracovávat synchronně v jedné transakci (limit 100 SOQL dotazů). Naplánování zpracování je zajištěno třídou `ProcessRequestsTasksBatch`, která implementuje `Batchable` rozhraní [54].

Pomocí tohoto rozhraní lze naplánovat asynchronní zpracování kolekce záznamů (úloh). Naplánování zpracování je znázorněno v kódu 3.13.

Kód 3.13: Předání úkolů ke zpracování (*ProcessRequestsTasksBatch.page*)

```
public class ProcessRequestsTasksBatch implements
    Database.Batchable<SObject>, Database.Stateful {
   private final Id requestId;
   private final static String NEW_VALUE = 'New';
   private final static String AUTOMATICALLY = 'Automatically';

   public ProcessRequestsTasksBatch(Id requestId) {
      this.requestId = requestId;
   }

   public Database.QueryLocator start(Database.BatchableContext
       bc) {
      return Database.getQueryLocator([SELECT Id FROM Task__c
           WHERE Request__c = :requestId AND Status__c =
               :NEW_VALUE AND Type__c = :AUTOMATICALLY]);
   }

   public void execute(Database.BatchableContext bc, List<Task__c>
       tasks) {
      UnitOfWork uow = new UnitOfWork();
      for (Task__c t : tasks) {
         try {
            PersonDataService pds = (PersonDataService)
                ClassUtils.getImplementation(
               PersonDataService.class.getName());
            pds.createIndividualsDataRecords(t.Id);
         } catch(Exception ex) {
            \\ error logging
         }
      }
      uow.commitWork();
   }

   public void finish(Database.BatchableContext bc) {
      \\ results logging
   }
}
```

Způsob zpracování úkolu je definován třídou `IndividualsDataParser`. Tato třída definuje, jakým způsobem budou k úkolu vygenerovány navázané `Individuals_Data__c` záznamy, jež obsahují sesbíraná data. Součástí systému jsou dvě třídy, pomocí kterých lze vyhledat data v Salesforce s využitím SOQL

dotazů.

Třída `IndividualsDataParser` slouží k vyhledání osobních údajů, jež jsou uloženy v jednotlivých atributech SObjectů (např. telefon, email). Třída `IndividualsDataParserMultiField` umožňuje vyhledávání osobních údajů na úrovni celých záznamů (např. objednávka, příležitost). U těchto záznamů je možné definovat, jaké atributy mají být použity dále ke zpracování.

Klient má možnost definovat vlastní způsob zpracování úkolů. Této vlastnosti může být využito například pro sběr dat pomocí API externích systémů.

3.2.5 Nastavení profilů

Pro nastavení oprávnění k jednotlivým funkcionalitám systému byly využity profily [24]. Pomocí profilů byl nastaven přístup k jednotlivým objektům, atributům, stránkám, třídám a zbylým částem systémů. Pro každou skupinu uživatelů definovanou v kapitole 2.2 byl vytvořen jeden profil.

Profily *GDPR Standard User* a *GDPR Request Handler User* byly vytvořeny naklonováním standardního profilu *Standard User*.

Profil GDPR Administrator byl vytvořen naklonováním standardního profilu *System Administrator*. Díky tomu bude mít administrátor oprávnění pro provádění úprav implementace systému.

Rozšíření oprávnění

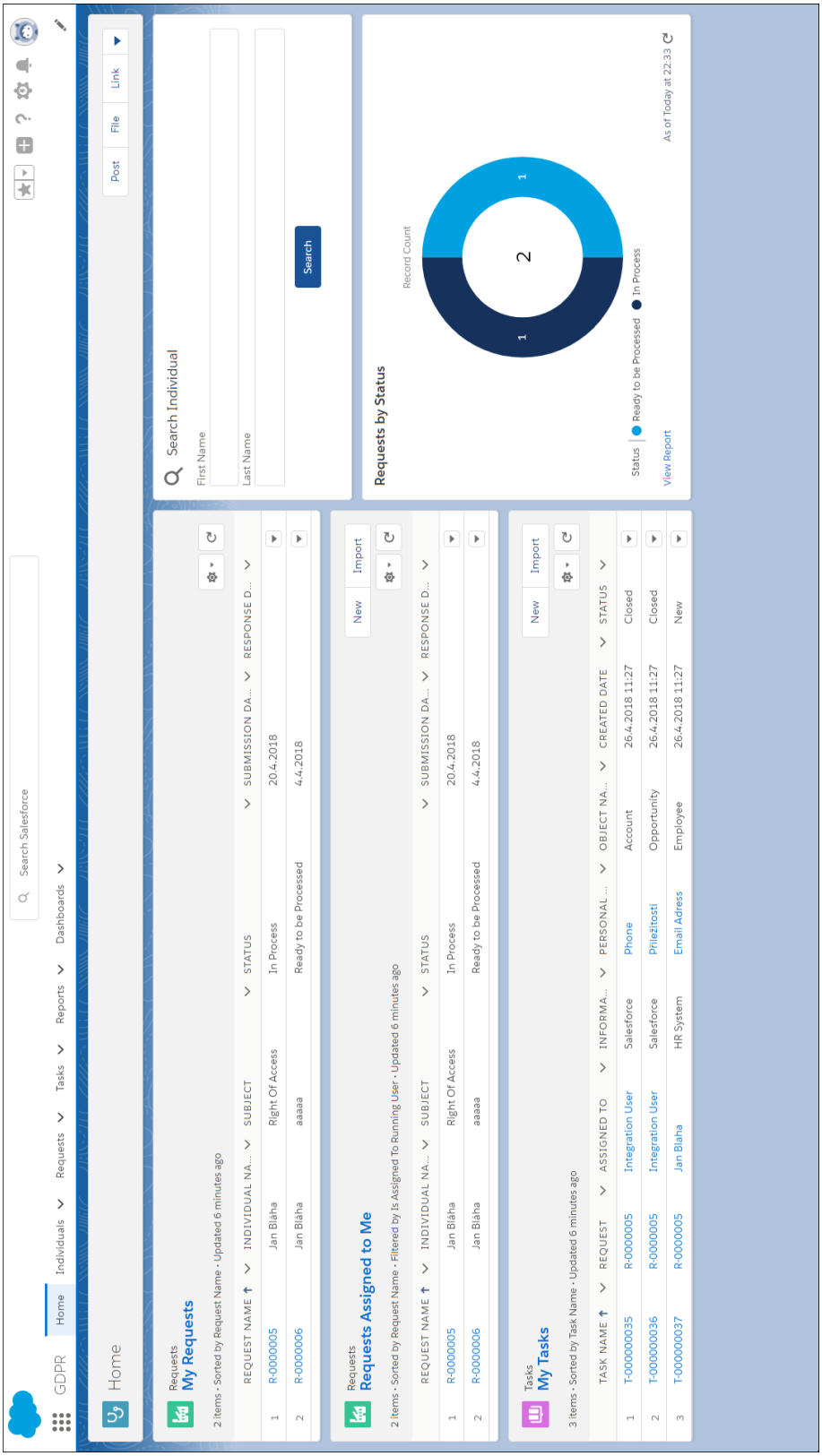
V případě, že by bylo potřeba rozšířit oprávnění některým uživatelů, můžou k tomu být použity *Permission Sets* [25]. Pomocí nich lze přidat práva pouze některým uživatelům, jelikož vztah mezi objekty *Permission Set* a *User* je ve vztahu M:N.

Tímto mechanismem lze rozšířit sadu práv uživatele, ale nelze jej využít pro jejich omezení.

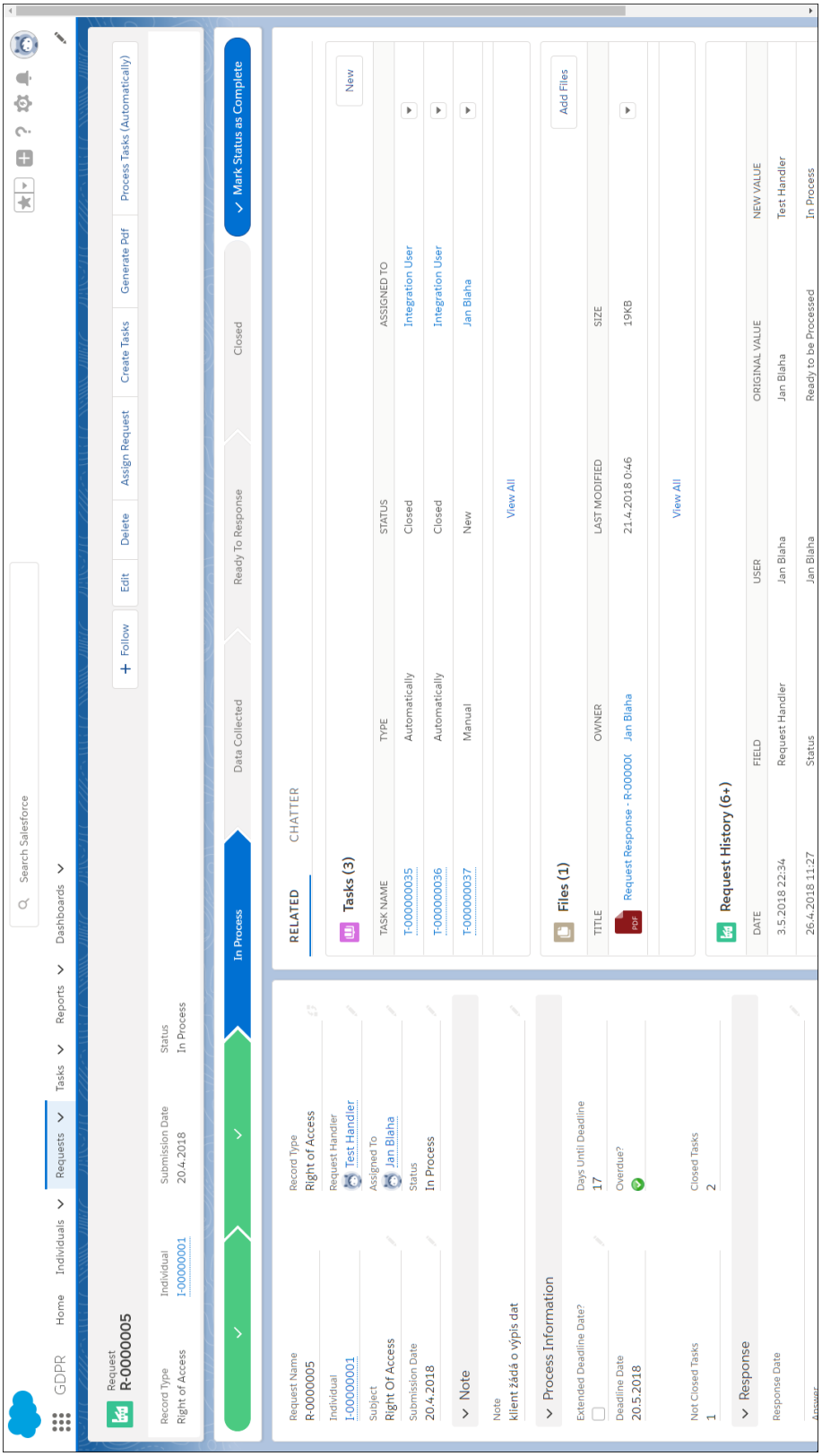
3.3 Ukázka aplikace

V této části jsou přiloženy náhledy naimplementovaného systému GDPR Podatelna. Uvedené snímky byly pořízeny po provedení uživatelského testování a zapracování jeho poznatků. Testování aplikace se věnuje následující kapitola 4.

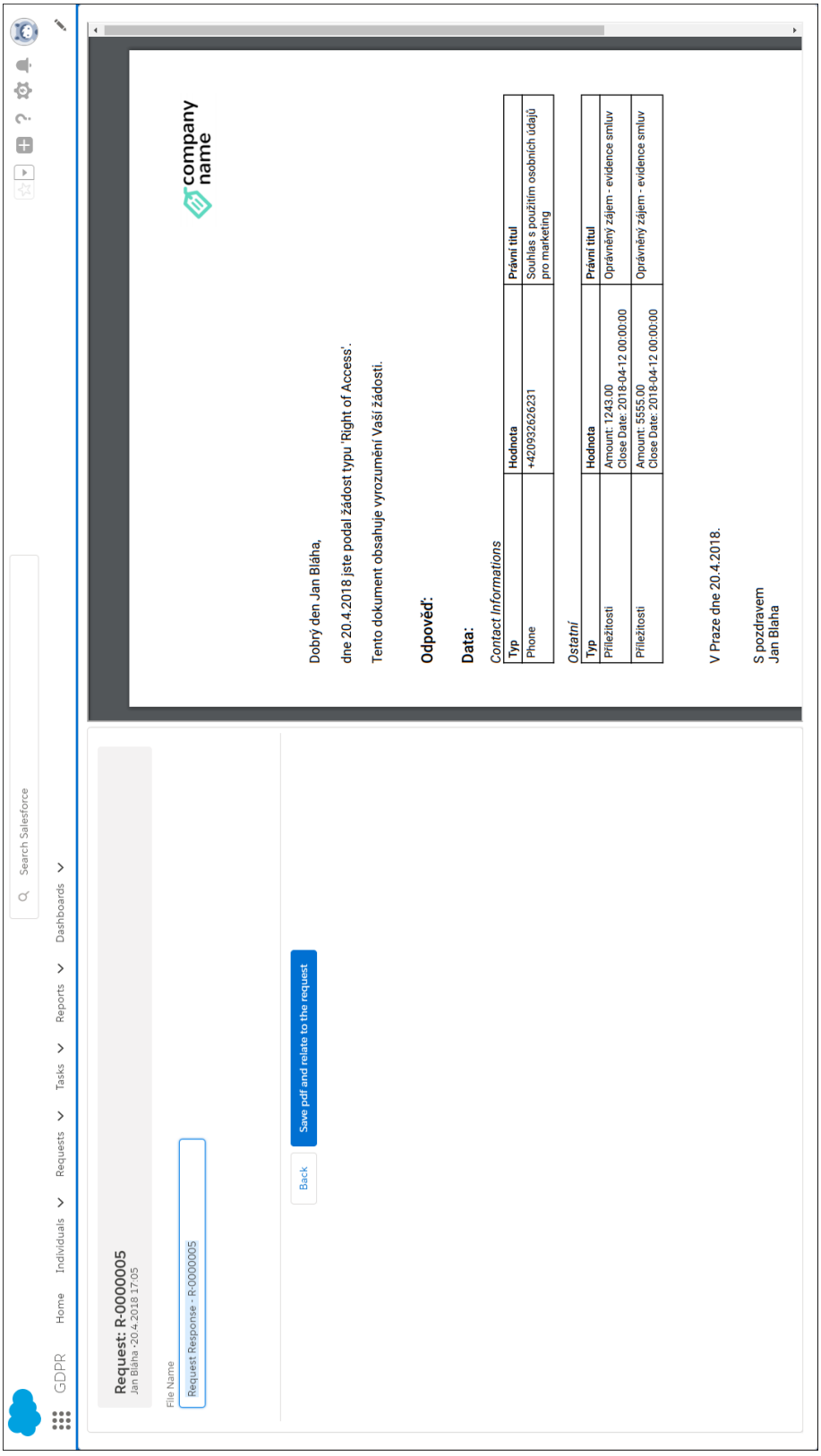
Na obrázku 3.2 je zobrazena domovská stránka aplikace. Obrázek 3.3 zobrazuje stránku detailu záznamu podané žádosti. Náhled stránky pro generování odpovědi na žádost subjektu ve podobě PDF dokumentu je zobrazen na obrázku 3.4.



Obrázek 3.2: Domovská stránka aplikace



Obrázek 3.3: Stránka záznamu žádosti subjektu



Obrázek 3.4: Stránka pro generování PDF

Kapitola 4

Testování

Tato kapitola obsahuje popis metodiky, jež byla použita pro otestování funkcionality naimplementovaného systému. Testování a ověření funkcionality kódu, jenž byl naprogramován v jazyce Apex, bylo realizováno pomocí jednotkových testů. Pro otestování uživatelského rozhraní výsledného systému bylo provedeno uživatelské testování reálným uživatelem dle předpřipravených testovacích scénářů.

4.1 Jednotkové testy

Při vývoji jednotlivých Apex tříd a triggerů byly souběžně implementovány příslušné jednotkové testy. Pro každou Apex třídu a trigger byla naimplementována jedna testovací třída. Každá testovací třída se skládá z několika testovacích metod, jež ověřují funkcionalitu tříd na úrovni jednotlivých metod.

Kód 4.1 obsahuje ukázku testovací metody, jež testuje funkcionalitu třídy *TaskTrigger_Helper*. Tato metoda testuje správné vyplnění atributů při vkládání záznamů objektu `Task__c`. Testování je provedeno pro 201 záznamů, aby bylo ověřeno, že nedochází k porušení limitů platformy pro SOQL dotazy a nedojde tak k vyhození výjimku a pádu programu. Test využívá pro přípravu dat třídu *TestFactory*, jež je popsána v následující části.

Třída TestFactory

Pro vytváření testovacích dat v jednotkových testech byla použita třída *TestFactory*. Tato třída vznikla upravením kódu třídy vyvinuté Danielem Hoechstem, jež je volně přístupná k použití pod licencí MIT [55].

Pomocí této třídy lze jednoduše definovat defaultní hodnoty jednotlivých SObjectů. Třída zároveň poskytuje metody pro získání instancí SObjectů a jejich vložení do databáze. Kód 4.1 obsahuje ukázku použití této třídy pro vytvoření testovacích dat. Způsob definice defaultních hodnot v třídě *TestFactory* je zobrazen v kódu 4.2.

Kód 4.1: Jednotkový test (*TaskTriggerHelper_Test.cls*)

```
@IsTest
public class TaskTriggerHelper_Test {

   @IsTest
   private static void testFillTypeAndOwnerInBulk() {
      // preparing related Personal_Data__c
      Information_System__c informationSystem =
          (Information_System__c) TestFactory.createSObject(new
          Information_System__c());
      informationSystem.Process_Automatically__c = true;
      insert informationSystem;

      Object__c obj = (Object__c) TestFactory.createSObject(new
          Object__c(Information_System__c = informationSystem.Id),
          true);

      Personal_Data__c personalData = (Personal_Data__c)
          TestFactory.createSObject(new Personal_Data__c());
      personalData.Object__c = obj.Id;
      insert personalData;

      Request__c request = (Request__c)
          TestFactory.createSObject(new Request__c(), true);

      // preparing tasks to be processed
      List<Task__c> taskList = new List<Task__c> ();
      for (Integer i = 0; i< 201; i++) {
         taskList.add(new Task__c(Request__c = request.Id,
             Personal_Data__c = personalData.Id, Type__c = null));
      }

      Test.startTest();
      insert taskList;
      Test.stopTest();

      // assertions
      taskList = [SELECT Type__c, Assigned_To__c FROM Task__c WHERE
          Id IN :taskList];
      for (Integer i = 0; i< 201; i++) {
         System.assertEquals('Automatically', taskList[i].Type__c);
         System.assertEquals(UserInfo.getUserId(),
             taskList[i].Assigned_To__c);
      }
   }
}
```

Kód 4.2: Definice defaultních hodnot (*TestFactory.cls*)

```
public class TaskDefaults implements FieldDefaults {
   public Map<Schema.SObjectField, Object> getFieldDefaults() {
      Personal_Data__c personalData = (Personal_Data__c)
          TestFactory.createSObject(new Personal_Data__c(), true);
      Request__c request = (Request__c)
          TestFactory.createSObject(new Request__c(), true);

      return new Map<Schema.SObjectField, Object> {
         Task__c.Personal_Data__c => personalData.Id,
         Task__c.Request__c => request.Id,
         Task__c.Status__c => 'New',
         Task__c.Type__c => 'Automatically'
      };
   }
}
```

Výsledek testování

Jak lze vidět na obrázku 4.1, všechny naimplementované jednotkové testy proběhly úspěšně. Tyto testy pokrývají 100 % řádků všech Apex tříd a triggerů v implementovaném systému. Pokrytí jednotlivých tříd je zobrazeno na obrázku 4.2.

✔	[View] AssignRequestController_Test	(5/5) Test Methods Passed
✔	[View] ClassUtils_Test	(5/5) Test Methods Passed
✔	[View] CreateTasksController_Test	(4/4) Test Methods Passed
✔	[View] EmailTemplateLoader_Test	(2/2) Test Methods Passed
✔	[View] GeneratePdfDefinitionServiceDefault_Test	(5/5) Test Methods Passed
✔	[View] GenerateRequestPdfController_Test	(6/6) Test Methods Passed
✔	[View] IndividualsDataParserMultiField_Test	(2/2) Test Methods Passed
✔	[View] IndividualsDataParser_Test	(4/4) Test Methods Passed
✔	[View] Logger_Test	(5/5) Test Methods Passed
✔	[View] PersonDataServiceDefault_Test	(2/2) Test Methods Passed
✔	[View] ProcessRequestsTasksBatch_Test	(1/1) Test Methods Passed
✔	[View] ProcessTasksController_Test	(3/3) Test Methods Passed
✔	[View] RenderElementStrategyImpls_Test	(7/7) Test Methods Passed
✔	[View] SearchIndividualController_Test	(4/4) Test Methods Passed
✔	[View] TaskServiceDefault_Test	(1/1) Test Methods Passed
✔	[View] TaskTriggerHelper_Test	(3/3) Test Methods Passed
✔	[View] TemplateParser_Test	(3/3) Test Methods Passed
✔	[View] TestFactory	(0/0) Test Methods Passed
✔	[View] UnitOfWork_Test	(5/5) Test Methods Passed

Obrázek 4.1: Protokol testování

Overall Code Coverage	
Class	Percent
Overall	**100%**
AssignRequestController	100%
ClassUtils	100%
CreateTasksController	100%
EmailTemplateLoader	100%
GeneratePdfDefinitionServiceDefault	100%
GenerateRequestPdfController	100%
IndividualsDataParser	100%
IndividualsDataParserMultiField	100%
Logger	100%
PersonDataServiceDefault	100%
ProcessRequestsTasksBatch	100%
ProcessTasksController	100%
RenderElementStrategyImpls	100%
SearchIndividualController	100%
TaskServiceDefault	100%
TaskTrigger	100%
TaskTrigger_Helper	100%
TemplateParser	100%
UnitOfWork	100%

Obrázek 4.2: Pokrytí jednotkových testů

4.2 Uživatelské testování

Pro ověření použitelnosti naimplementovaného systému bylo provedeno uživatelské testování reálným uživatelem. Testování bylo provedeno na hotové aplikaci. Pro testování byly vybrány dva stěžejní případy užití systému (UC6 a UC31).

4.2.1 Testovací scénáře

1) Založení žádosti

- Zadání:

  Vyhledejte v systému subjekt se jménem „Jan Novák“ a založte pro něj novou žádost typu „Right of Access“. K vytvořené žádosti nahrajte libovolný soubor jako přílohu.

- Předpoklady:

  Testovací scénář předpokládá správné nakonfigurování metadat osobních údajů a nastavení práv, jež jsou potřeba pro vykonání akce. Dále scénář předpokládá existenci záznamu subjektu v databázi.

- Očekávaný scénář:

  1. Uživatel se přesune na domovskou stránku aplikace.
  2. Uživatel se přesune do formuláře pro vyhledávání subjektů.
  3. Uživatel zadá hodnotu „Jan“ do pole „First Name“.
  4. Uživatel zadá hodnotu „Novák“ do pole „Last Name“.
  5. Uživatel stiskne tlačítko „Search“.
  6. Uživatel se pomocí odkazu ve výsledcích přesune na záznam subjektu.
  7. Uživatel v seznamu žádostí klikne na tlačítko „“New.
  8. Uživatel zvolí typ žádosti „Right of Access“ a stiskne tlačítko „Next“.
  9. Uživatel zadá hodnotu „Žádost o přístup“ do pole „Subject“.
  10. Uživatel do pole „Submission Date“ vyplní aktuální datum.
  11. Uživatel nastaví status na „Collecting Information from Individual“
  12. Uživatel stiskne tlačítko „Save“.
  13. Uživatel v seznamu příloh stiskne tlačítko „Add Files“.
  14. V otevřeném formuláři uživatel stiskne tlačítko „Upload Files“, zvolí přílohu na disku a stiskne tlačítko „Done“.

- Očekávaný výsledek:

  Uživatel je na obrazovce žádosti a vidí v seznamu příloh nahraný soubor.

2) Vygenerování PDF dokumentu

- Zadání:

  Pro žádost „R-0000624“ vygenerujte PDF dokument s odpovědí. Vygenerovaný dokument uložte k záznamu žádosti.

- Předpoklady:

  Testovací scénář předpokládá správné nakonfigurování metadat osobních údajů a nastavení práv, jež jsou potřeba pro vykonání akce. Dále scénář předpokládá existenci záznamu zpracované žádosti „R-0000624“ s vyplněnými údaji.

- Očekávaný scénář:
  1. Uživatel se přesune na domovskou stránku aplikace.
  2. Uživatel v seznamu „My Requests“ zvolí žádost „R-0000624“ a přesune se na její detail.
  3. Uživatel mezi akcemi dostupnými na detailu záznamu stiskne tlačítko „Generate PDF“.
  4. Uživatel zadá hodnotu „Odpověď na žádost R-0000624“ do pole „File Name“.
  5. Uživatel klikne na tlačítko „Save and relate to the request“.
  6. Uživatel klikne na tlačítko „Back“ a vrátí se na stránku záznamu žádosti.
- Očekávaný výsledek:

  Uživatel je na obrazovce záznamu žádosti „R-0000624“ a mezi přílohami vidí vygenerovaný dokument.

4.2.2 Průběh testování

Testovací scénáře byly provedeny uživatelem a byl sledován průběh provádění vzhledem k očekávaným scénářům. Po dokončení scénáře byl uživatel požádán o zpětnou vazbu.

Charakteristika uživatele

- 25 let
- student vysoké školy se zaměřením na IT
- pokročilý uživatel
- seznámen s problematikou systému

Scenář 1

Uživatel provedl zadání podle očekávaného scénáře. Na žádném kroku scénáře se neobjevily delší prodlevy nebo zaváhání. Uživatel navrhuje přidat defaultní hodnotu pro pole „Status“ na záznamu žádosti, jelikož v drtivé většině případů bude jeho hodnota při zakládání žádosti stejná.

Scenář 2

Uživatel neměl s provedením scénáře žádný problém. Test byl proveden podle očekávaného scénáře. Stejně jako u předchozího scénáře, uživatel navrhuje přidat defaultní hodnotu pro název vygenerovaného dokumentu, jež by mohla být případně změněna.

4.2.3 Zhodnocení testování

Během uživatelského testování nebyly objeveny žádné kritické nedostatky. Uživatel navrhl dvě vylepšení uživatelského rozhraní, konkrétně vyplňování defaultních hodnot při zakládání nových žádostí a generování dokumentů. Navržené změny byly zapracovány a jednotlivé scénáře byly opět otestovány uživatelem. Při opětovném testování již nedošlo k žádným připomínkám ze strany uživatele.

Závěr

Hlavním cílem této práce bylo navrhnout a naimplementovat systém GDPR Podatelna na platformě Salesforce, který bude podporovat proces sběru a zpracování žádostí, jež mohou občané nově podávat na základě nařízení GDPR.

V první části práce je představena problematika ochrany osobních údajů, jež je nově regulována nařízením GDPR. Jsou zde shrnuty nová práva a povinnosti občanů a institucí podle nařízením GDPR. V závěru kapitoly jsou představena řešení dostupná na našem trhu zabývající se touto problematikou.

Po seznámení se s nařízením GDPR byla provedena analýza a návrh řešení systému pro zpracování jednotlivých typů žádostí. V první fázi byly stanoveny funkční a nefunkční požadavky kladené na systém. Pro každou roli uživatelů byla definována sada případů užití, které realizují jednotlivé požadavky. Pro definici entit v systému byl vytvořen doménový model. Součástí této fáze byl také návrh uživatelského rozhraní pomocí wireframes. V poslední části kapitoly je stručně popsána cloudová platforma Salesforce společně se specifiky, jež ovlivnily návrh systému.

Po analýze a návrhu řešení systému GDPR Podatelna byla provedena jeho implementace. Ve třetí kapitole jsou popsány technologie platformy Salesforce, jež byly využity k implementaci. Dále jsou zde popsány detaily implementace požadavků, při kterých bylo použito atypických nástrojů nebo postupů. V závěru kapitoly je přiložena ukázka výsledného systému.

V poslední kapitole je popsána metodika testování, pomocí které byla ověřena funkčnost systému. Testování bylo realizováno pomocí jednotkových a uživatelských testů.

Výstupem práce je aplikace na platformě Salesforce pro zpracování žádostí, jež mohou občané podávat na základě nařízení GDPR. Naimplementovaný systém splňuje požadavky vymezené během analýzy a byl řádně otestován. Tím byly úspěšně splněny všechny body zadání diplomové práce.

Při psaní této práce jsem se seznámil s problematikou ochrany osobních údajů a nařízením GDPR. Zároveň jsem si rozšířil své znalosti nástrojů platformy Salesforce.

Možnosti rozšíření

Systém byl navržen tak, aby byl v budoucnu snadno rozšiřitelný, a aby ho bylo možné upravovat pro potřeby jednotlivých klientů. Pro reálné nasazení je tedy systém potřeba individuálně upravit a případně rozšířit, což je v souladu s požadavkem na rozšiřitelnost.

Pro zpracování žádostí typu „Právo na přístup k osobním údajům“ by bylo vhodné doimplementovat automatizované sbírání dat z externích systému prostřednictvím poskytnutého rozhraní.

Generování dokumentů pro odpovědi žádostí je v současném systému definováno pouze na základě šablon. Systém by mohl být rozšířen o nástroj, pomocí kterého by zpracovatelé žádostí mohli sestavovat dokument pro jednotlivé žádosti individuálně.

Dalším možným rozšířením systému by mohla být automatizace zbylých typů žádostí. Tato rozšíření by byla pravděpodobně provedena v instanci konkrétního klienta, jelikož způsob zpracování konkrétního typu žádosti bude v každé společnosti jiný.

Literatura

[1] Evropský parlament a Rada Evropské unie: *Obecné nařízení o ochraně osobních údajů*. [cit. 2018-04-20]. Dostupné z: http://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1501688126470&uri=CELEX:32016R0679

[2] Český statistický úřad: *Informační technologie [online]*. [cit. 2018-04-20]. Dostupné z: https://www.czso.cz/csu/czso/informacni_technologie_pm

[3] Carole Cadwalladr, Emma Graham-Harrison (The Guardian): *Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach [online]*. [cit. 2018-04-20]. Dostupné z: https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election

[4] Sbírka zákonů České republiky: *Zákon o ochraně osobních údajů 101/2000 Sb.* [cit. 2018-04-20]. Dostupné z: http://www.cz-museums.cz/UserFiles/File/Legislativa/zakon-101-2000.pdf

[5] Týden.cz: *Brabec: Zákon k nařízení GDPR do května nestihneme schválit [online]*. [cit. 2018-04-20]. Dostupné z: https://www.tyden.cz/rubriky/domaci/politika/brabec-zakon-k-narizeni-gdpr-do-kvetna-nestihneme-schvalit_471090.html

[6] Risk Analysis Consultants: *ISMS: normy ISO 27001 a ISO 27002 [online]*. [cit. 2018-04-20]. Dostupné z: http://www.rac.cz/rac/homepage.nsf/CZ/BS7799

[7] Parlament České republiky: *Zákon č. 181/2014 Sb., Zákon o kybernetické bezpečnosti a o změně souvisejících zákonů (zákon o kybernetické bezpečnosti)*. [cit. 2018-04-20]. Dostupné z: https://www.zakonyprolidi.cz/cs/2014-181

[8] Information Commissioner's Office (ico): *Guide to the General Data Protection Regulation (GDPR) - Lawful basis for processing [online].* [cit. 2018-04-20]. Dostupné z: https://ico.org.uk/for-organisations/guide-to-the-general-data-protection-regulation-gdpr/lawful-basis-for-processing/

[9] GDPR.cz: *Jaké povinnosti ukládá GDPR institucím a firmám [online].* [cit. 2018-04-20]. Dostupné z: https://www.gdpr.cz/gdpr/povinnosti/

[10] O_2: *O_2 GDPR Partner - podrobná on-line analýza [online].* [cit. 2018-04-20]. Dostupné z: https://www.o2.cz/podnikatel/gdpr-vice-o-analyze/

[11] Centrum pro ochranu osobních údajů: *Ochrana osobních údajů [online].* [cit. 2018-04-20]. Dostupné z: https://www.ochranaosobnichudaju.cz/

[12] Centrum pro ochranu osobních údajů: *OCHRANOU [online].* [cit. 2018-04-20]. Dostupné z: https://www.ochranou.cz/

[13] xGDPR Express: *Informace o GDPR software [online].* [cit. 2018-04-20]. Dostupné z: https://www.xgdpr.com/cs-cz/Informace-o-GDPR-software-xGDPR-Express

[14] DataLite: *GDPR nástroj eDPO [online].* [cit. 2018-04-20]. Dostupné z: https://www.edpo.cz/

[15] eDPO - Průvodce GDPR od A do Z: *Licenční politika (ceník) [online].* [cit. 2018-04-20]. Dostupné z: https://www.edpo.cz/cenik.html

[16] OneTrust: *About OneTrust [online].* [cit. 2018-04-20]. Dostupné z: https://onetrust.com/company/about-us/

[17] OneTrust: *Pricing [online].* [cit. 2018-04-20]. Dostupné z: https://onetrust.com/pricing/

[18] Quality System Solutions: *DataPro Tools [online].* [cit. 2018-04-20]. Dostupné z: http://datapro.tools/background/

[19] AppExchange: *GDPR tools for Data Privacy Management: DataPro Tools [online].* [cit. 2018-04-20]. Dostupné z: https://appexchange.salesforce.com/appxListingDetail?listingId=a0N3A00000FAAYNUA5

[20] Salesforce.com: *AppExchange [online].* [cit. 2018-04-20]. Dostupné z: https://appexchange.salesforce.com/

[21] TechTerms: *PDF (Portable Document Format) Definitions [online].* [cit. 2018-04-20]. Dostupné z: https://techterms.com/definition/pdf

[22] FileInfo: *CSV File Extension [online]*. [cit. 2018-04-24]. Dostupné z: https://fileinfo.com/extension/csv

[23] Google: *Google Chrome [online]*. [cit. 2018-04-20]. Dostupné z: https://www.google.com/chrome/

[24] Salesforce Docs [online]: *Profiles*. [cit. 2018-04-24]. Dostupné z: https://help.salesforce.com/articleView?id=admin_userprofiles.htm

[25] Salesforce Docs: *Permission Sets [online]*. [cit. 2018-04-24]. Dostupné z: https://help.salesforce.com/articleView?id=perm_sets_overview.htm

[26] Salesforce Docs: *User Role Hierarchy [online]*. [cit. 2018-04-24]. Dostupné z: https://help.salesforce.com/articleView?id=admin_roles.htm

[27] Salesforce Docs: *Validation Rules [online]*. [cit. 2018-04-24]. Dostupné z: https://help.salesforce.com/articleView?id=fields_about_field_validation.htm

[28] Salesforce Docs: *Which Automation Tool Do I Use? [online]*. [cit. 2018-04-24]. Dostupné z: https://help.salesforce.com/articleView?id=process_which_tool.htm

[29] Balsamiq: *Balsamiq. Rapid, effective and fun wireframing software. [online]*. [cit. 2018-04-24]. Dostupné z: https://balsamiq.com/

[30] Salesforce EMEA: *What is Salesforce? Cloud CRM Solutions [online]*. [cit. 2018-04-24]. Dostupné z: https://www.salesforce.com/eu/products/what-is-salesforce/

[31] Investopedia: *Software As A Service (SaaS) [online]*. [cit. 2018-04-24]. Dostupné z: https://www.investopedia.com/terms/s/software-as-a-service-saas.asp

[32] Salesforce EMEA: *Small Business CRM [online]*. [cit. 2018-04-24]. Dostupné z: https://www.salesforce.com/eu/solutions/small-business-solutions/overview/?d=70130000002DpEY

[33] Salesforce.com: *Salesforce App: Mobile CRM Apps [online]*. [cit. 2018-04-24]. Dostupné z: https://www.salesforce.com/solutions/mobile/overview/

[34] Salesforce.com: *Editions and Pricing - Sales Cloud [online]*. [cit. 2018-04-24]. Dostupné z: https://www.salesforce.com/editions-pricing/sales-cloud/

[35] Salesforce.com: *GDPR Overview [online]*. [cit. 2018-04-24]. Dostupné z: https://www.salesforce.com/gdpr/overview/

[36] Salesforce Spring '18 Release Notes: *Store Certain Data Privacy Preferences [online]*. [cit. 2018-04-24]. Dostupné z: https://releasenotes.docs.salesforce.com/en-us/spring18/release-notes/rn_general_store_data_privacy_pref.htm

[37] Salesforce Docs: *Understanding Packages [online]*. [cit. 2018-04-28]. Dostupné z: https://help.salesforce.com/articleView?id=sharing_apps.htm

[38] Salesforce: *ISVforce Guide (Version 42.0, Spring '18)*. [cit. 2018-04-28]. Dostupné z: https://resources.docs.salesforce.com/212/latest/en-us/sfdc/pdf/salesforce_packaging_guide.pdf

[39] developer.force.com: *The Force.com Multitenant Architecture [online]*. [cit. 2018-04-24]. Dostupné z: https://developer.salesforce.com/page/Multi_Tenant_Architecture

[40] Salesforce Developers - Apex Developer Guide: *Execution Governors and Limits [online]*. [cit. 2018-04-28]. Dostupné z: https://developer.salesforce.com/docs/atlas.en-us.210.0.apexcode.meta/apexcode/apex_gov_limits.htm

[41] Techopedia: *What is Java Bytecode? [online]*. [cit. 2018-04-28]. Dostupné z: https://www.techopedia.com/definition/7866/java-bytecode

[42] Salesforce Docs: *Apex Developer Guide - Differences Between Apex Classes and Java Classes [online]*. [cit. 2018-04-28]. Dostupné z: https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/apex_classes_java_diffs.htm

[43] Salesforce Developers - Visualforce Developer Guide: *What is Visualforce? [online]*. [cit. 2018-04-28]. Dostupné z: https://developer.salesforce.com/docs/atlas.en-us.pages.meta/pages/pages_intro_what_is_it.htm

[44] Salesforce Docs: *Customize Detail Page Buttons [online]*. [cit. 2018-04-28]. Dostupné z: https://help.salesforce.com/articleView?id=customizing_detail_page_buttons.htm

[45] Salesforce Docs - Lightning Components Developer Guide: *Why Use the Lightning Component Framework? [online]*. [cit. 2018-04-28]. Dostupné z: https://developer.salesforce.com/docs/atlas.en-us.lightning.meta/lightning/intro_benefits.htm

[46] Salesforce: *Lightning Design System [online]*. [cit. 2018-04-28]. Dostupné z: https://www.lightningdesignsystem.com/

[47] Salesforce - Lightning Design System: *Component Overview [online]*. [cit. 2018-04-28]. Dostupné z: https://www.lightningdesignsystem.com/components/overview/

[48] The jQuery Foundation: *jQuery [online]*. [cit. 2018-05-02]. Dostupné z: http://jquery.com/

[49] Bartek Pampuch: *Pdfmake [online]*. [cit. 2018-05-02]. Dostupné z: http://pdfmake.org/

[50] Bartek Pampuch: *Pdfmake License [online]*. [cit. 2018-05-02]. Dostupné z: https://github.com/bpampuch/pdfmake/blob/master/LICENSE

[51] Martin Fowler: *Inversion of Control Containers and the Dependency Injection pattern [online]*. [cit. 2018-05-02]. Dostupné z: https://martinfowler.com/articles/injection.html

[52] Salesforce Docs - Apex Developer Guide: *Type Class [online]*. [cit. 2018-05-02]. Dostupné z: https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/apex_methods_system_type.htm

[53] Salesforce Docs: *Email Templates [online]*. [cit. 2018-05-02]. Dostupné z: https://help.salesforce.com/articleView?id=email_templates_landing_page.htm

[54] Salesforce Docs - Apex Developer Guide: *Batchable Interface [online]*. [cit. 2018-05-02]. Dostupné z: https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/apex_interface_database_batchable.htm

[55] Daniel Hoechst: *Salesforce-Test-Factory [online]*. [cit. 2018-05-02]. Dostupné z: https://github.com/dhoechst/Salesforce-Test-Factory

Příloha **A**

Seznam použitých zkratek

API Application Programming Interface

CRM Customer Relationship Management

CRUD Create, Read, Update, Delete

CSS Cascading Style Sheets

CSV Comma-separated values

DML Data manipulation language

DOM Document Object Model

DPO Data Protection Officer

GDPR General Data Protection Regulation

GUI Graphical User Interface

HTML HyperText Markup Language

IoT Internet of Things

IS informační systém

ISO International Organization for Standardization

PDF Portable Document Format

SLDS Salesforce Lightning Design System

SOQL Salesforce Object Query Language

ÚOOÚ Úřad pro ochranu osobních údajů

XML Extensible Markup Language

Příloha **B**

Obsah přiloženého CD

- `readme.txt` ..... stručný popis obsahu CD
- `src`
  - `impl` ..... zdrojové kódy implementace
  - `thesis` ..... zdrojová forma práce ve formátu LaTeX
- `text` ..... text práce
  - `thesis.pdf` ..... text práce ve formátu PDF