



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ

DEPARTMENT OF INFORMATION SYSTEMS

**SYSTÉM PRO AGREGACI INFORMACÍ Z VEŘEJNÝCH
DATABÁZÍ**

SYSTEM FOR AGREGATION OF INFORMATION FROM PUBLIC DATABASES

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. JAKUB POJSL

VEDOUcí PRÁCE

SUPERVISOR

Mgr. Ing. PAVEL OČENÁŠEK, Ph.D.

BRNO 2023

Zadání diplomové práce



147068

Ústav: Ústav informačních systémů (UIFS)
Student: **Pojsl Jakub, Bc.**
Program: Informační technologie a umělá inteligence
Specializace: Softwarové inženýrství
Název: **Systém pro agregaci informací z veřejných databází**
Kategorie: Web
Akademický rok: 2022/23

Zadání:

1. Seznamte se s existujícími aplikacemi a možnostmi automatizace procesů advokátní kanceláře (AK). Seznamte se s možnostmi získávání informací prostřednictvím internetu (eJustice) a z veřejně dostupných elektronických rejstříků.
2. Po konzultaci s vedoucím práce analyzujte požadavky na systém, který bude vést spisy AK, bude automaticky získávat aktuální informace o řízení a účastnících z internetu a vhodným způsobem získaná data prezentovat uživateli. Systém bude zahrnovat správu spisů, správu dokumentů, evidenci pošty. Dále bude podporovat generování podkladů pro účtování provedené práce klientům AK.
3. Na základě výsledků analýzy navrhnete vlastní systém, který bude rozšiřovat funkce již existujících systémů a usnadní automatizaci procesů v konkrétní IK
4. Navržený systém implementujte a ověřte na reálných datech.
5. Diskutujte vlastní přínos a odlišnosti od již existujících systémů a navrhnete možnosti dalšího rozšíření.

Literatura:

- Freeman, A., Sanderson, S. *Pro ASP.NET MVC 3 framework*, New York: Apress, 2011. ISBN: 978-1-4302-3404-3.
- Budiš, P., Štědroň, B. *Elektronická komunikace (Electronic communication)*. Bratislava: Magnet Press, 2008. ISBN 978-80-89169-11-5.
- Arlow, J. *UML a unifikovaný proces vývoje aplikací*. Brno: Computer Press, 2003. ISBN 80-7226-947-X.
- Welling, L., Thomsonová, L. *PHP a MySQL - rozvoj webových aplikací - 3. vydání*, Praha: Softpress, 2005. ISBN 80-86497-83-6.

Při obhajobě semestrální části projektu je požadováno:

Body 1 - 3 zadání.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Očenášek Pavel, Mgr. Ing., Ph.D.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1.11.2022
Termín pro odevzdání: 17.5.2023
Datum schválení: 28.10.2022

Abstrakt

Práce se zabývá návrhem a implementací systému pro advokátní kanceláře, který automaticky získává relevantní informace z veřejných databází. Na úvod je čtenář seznámen s důležitostí a přístupností vybraných veřejných rejstříků v kontextu platné legislativy. Dále je provedena analýza procesů advokátní kanceláře a aktuálně používaných informačních systémů v této oblasti. Práce se dále zabývá specifikací a návrhem systému, který zahrnuje funkce pro správu klientů, jejich spisů a dokumentů, správu pošty a ulehčení účtování vykonané práce klientům. Praktickou část tvoří implementace systému s důrazem na aspekty uživatelského rozhraní, zpřístupnění dat z externích zdrojů a zefektivnění procesů advokátní kanceláře. Výsledkem práce je kompletní systém pro podporu agendy advokátní kanceláře s agregací dat z vybraných veřejných databází, sledováním změn soudních a insolvenčních řízení a s napojením na systém datových schránek. Součástí je také zveřejněné rozhraní pro získání agregovaných dat. Systém byl nasazený na testovací server, otevírá možnosti pro další rozšíření a má potenciál reálného využití v praxi.

Abstract

The thesis focuses on the design and implementation of a system for law firms, designed to automatically retrieve relevant information from public databases. Initially, the reader is familiarized with the significance and accessibility of chosen public registers within the context of prevailing legislation. This is followed by an analysis of the processes within law firms and the existing information systems in their area. Subsequently, the focus shifts to the detailed specification and design of this system, encapsulating functionalities for client management, document organization, mail handling, and simplified billing procedures. The practical aspect of the thesis includes the actual implementation of the system, with a particular focus on user interface design, data accessibility from external sources, and the streamlining of law firm processes. The outcome of this work is a comprehensive system to aid the law firm's operations. This includes aggregating data from selected public databases, monitoring changes in court and insolvency proceedings, and integrating with a government electronic mailbox system. Additionally, the system offers a publicly available interface for obtaining aggregated data from public registers. The system has been deployed in a testing environment, paving the way for further enhancements and demonstrating the potential for real-world application.

Klíčová slova

Advokátní kanceláře, Získávání informací, Webová aplikace, Agregace dat, Veřejné databáze, Veřejné rejstříky, Správa klientů, Správa spisů, Účtování práce, Fakturace, UI

Keywords

Law firms, Information retrieval, Web application, Data Aggregation, Public databases, Public registers, Client Management, File management, Work invoicing, Billing, UI

Citace

POJSL, Jakub. *Systém pro agregaci informací z veřejných databází*. Brno, 2023. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Mgr. Ing. Pavel Očenášek, Ph.D.

System pro agregaci informaci z veřejných databází

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Mgr. Ing. Pavla Očenáška, Ph.D. Další informace mi poskytly osoby se zkušenostmi s právní praxí. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Jakub Pojsl
17. května 2023

Poděkování

Rád bych vyjádřil upřímné poděkování své přítelkyni Natálii za nekonečnou podporu během mého zpracování této práce. Dále bych chtěl velmi poděkovat svému vedoucímu Mgr. Ing. Pavlu Očenáškoví, Ph.D. za cenné rady a vedení. Taktéž bych rád poděkoval všem osobám z právní praxe, které mi ochotně poskytly zpětnou vazbu k vytvořenému systému.

Obsah

1	Úvod	2
2	Veřejné databáze a agenda advokátní kanceláře	3
2.1	Veřejné rejstříky	3
2.2	Informace o soudních řízeních	7
2.3	Procesy v právní kanceláři	9
3	Informační systémy a existující řešení	16
3.1	Tvorba informačních systémů	16
3.2	Systémy využívané v právních kancelářích	20
3.3	Vybrané implementační technologie	27
4	Specifikace a návrh systému	29
4.1	Specifikace požadavků na systém	29
4.2	Napojení na veřejné databáze	34
4.3	Datový model a návrh databáze	36
4.4	Funkcionalita a uživatelské rozhraní	40
4.5	Návrh architektury systému	43
5	Implementace systému pro právní kanceláře	47
5.1	Struktura projektu	47
5.2	Uživatelské účty a zabezpečení	51
5.3	Získávání a agregace dat z veřejných databází	54
5.4	Nahrávání a správa dokumentů	60
5.5	Správa pošty a datové schránky	66
5.6	Vykazování a účtování vykonané práce	70
6	Testování a nasazení	75
6.1	Testování funkcionality	75
6.2	Zveřejnění implementovaného systému	77
6.3	Uživatelské testování	77
7	Závěr	78
7.1	Shrnutí dosažených výsledků	78
7.2	Možná budoucí rozšíření	79
	Literatura	81

Kapitola 1

Úvod

Cílem práce je vytvořit komplexní systém sloužící advokátním kancelářím pro přehledné vedení jejich každodenní agendy, který bude zároveň umožňovat zefektivnění některých procesů díky automatickému získávání informací z veřejně dostupných zdrojů a databází. Práce se zabývá podrobnou analýzou veřejných databází a existujících řešení pro advokátní kanceláře a sestavením specifikací požadavků na vytvářený systém. Na základě těch je systém navržen, implementován a otestován.

V první části práce jsou popsány dostupné elektronické zdroje informací relevantní pro advokáty a právní kanceláře, jako jsou vybrané veřejné rejstříky a portál eJustice provozovaný Ministerstvem spravedlnosti České republiky. Poté je provedena analýza a popis nejčastějších procesů advokátní kanceláře na základě poznatků získaných pozorováním v konkrétní advokátní kanceláři, na základě rozhovorů a spolupráce s několika zaměstnanci advokátní kanceláře a studia relevantní literatury týkající se advokátního práva.

Následuje kapitola zabývající se úvodem do tvorby informačních systémů a jejich architektury. Na tento úvod je poté navázáno podrobnou analýzou existujících řešení a informačních systémů pro právní kanceláře. Je proveden popis jimi nabízené funkcionality a dalších klíčových vlastností. V závěru této části jsou popsány vybrané implementační technologie.

Druhá část práce se zabývá sběrem požadavků na vytvářený systém a sestavením podrobného souboru specifikací požadavků, který slouží jako základ pro následný návrh systému a jeho výslednou implementaci. Na základě specifikovaných požadavků na systém je vytvořen detailní návrh systému zahrnující vytvoření hlavních návrhových diagramů s využitím jazyka UML, jako například ER diagram a návrh architektury celého systému. Poté je přiblížen průběh iterativního návrhu uživatelského rozhraní. Tento návrh zároveň sloužil pro získání zpětné vazby od cílových uživatelů již při návrhu systému. Vývoj systému byl během návrhu rozdělen do 3 iterací z nichž třetí je považována za pokračování této diplomové práce.

Ve čtvrté kapitole je rozebrán postup vývoje samotného systému a jsou zde blíže popsány vybrané implementační části, jako jsou autentizace, modul pro agregaci dat či správa dokumentů a modul pro účtování práce. Předposlední kapitola se zabývá testováním celého systému, jeho nasazením do testovacího prostředí a následným uživatelským testováním se zaměstnanci advokátní kanceláře. Na závěr je provedeno vyhodnocení dosažených výsledků a jsou nastíněny možnosti dalšího rozšiřování funkcionality.

Kapitola 2

Veřejné databáze a agenda advokátní kanceláře

V této kapitole jsou analyzovány informace nezbytné pro úspěšné vytvoření požadovaného systému. Cílem je získat co nejpřesnější informace o fungování advokátních kanceláří a pochopit jejich vnitřní procesy. Kapitola se zaměřuje zejména na povahu dat, s kterými právní kanceláře pracují a na způsob jejich organizace. Neopomíná veřejné rejstříky, databáze a další online zdroje informací relevantních pro advokáty a advokátní kanceláře. Nastihuje způsoby, jakými jsou data z těchto zdrojů využívána, jaká je jejich struktura a jak k nim zaměstnanci advokátních kanceláří či advokáti přistupují. Dále kapitola přibližuje jakými způsoby je možné tato data získat automatizovaně. Na základě pozorování jsou detailně popsány vybrané procesy právní kanceláře s návazností na využití informačních technologií.

2.1 Veřejné rejstříky

Veřejné rejstříky jsou v České republice upraveny zákonem č. 89/2012 Sb., občanský zákoník [37] a zákonem č. 304/2013 Sb., zákon o veřejných rejstřících právnických a fyzických osob a o evidenci svěřenských fondů [41]. právní úprava v občanském zákoníku je zakotvena v § 120 a § 121. Druhý ze zmiňovaných zákonů lze ve vztahu k občanskému zákoníku vnímat jako *lex specialis*. Je vhodné zmínit rozdíl mezi pojmem veřejný rejstřík a veřejný seznam dle občanského zákoníku. **Veřejným rejstřikem** je rejstřík, do kterého se zaznamenávají informace o osobách, a to jak právnických, tak fyzických, stanovené zákonem. Naopak **veřejný seznam** je seznam, do kterého se zapisují informace o věcech předpokládané zákonem. Příkladem veřejného seznamu je katastr nemovitostí, zatímco obchodní rejstřík je veřejným rejstřikem. Je vhodné také dodat, že pojem veřejný neznamena, že každý rejstřík nebo seznam je veřejně přístupný pro všechny. To, zda je určitý seznam nebo rejstřík veřejný stanoví konkrétní zákon. Registr vozidel je příkladem “neveřejného” seznamu. [47]

Dle § 1 zákona o veřejných rejstřících právnických a fyzických osob a o evidenci svěřenských fondů, jsou veřejnými rejstřiky následující:

- spolkový rejstřík,
- nadační rejstřík,
- rejstřík ústavů,
- rejstřík společenství vlastníků jednotek,

- obchodní rejstřík,
- rejstřík obecně prospěšných společností. [41]

Stejnomené ustanovení v třetím odstavci uvádí, že veřejný rejstřík je **informačním systémem veřejné správy**. Vzhledem k postupující digitalizaci v různých sektorech, včetně státního, ustanovení stanovuje jasné pravidlo, podle kterého je veřejný rejstřík veden v **elektronické podobě** a je veden tzv. **rejstříkovým soudem** (ten do něj zapisuje údaje stanovené zákonem a také je může mazat). Úkolem rejstříkových soudů je zajistit soulad zapisovaných údajů se skutečným stavem. Rejstříkovým soudem je krajský soud podle místa, v jehož obvodu se nachází obecný soud osoby, jejíž údaje se do rejstříku zapisují. [47]

S pojem veřejný rejstřík úzce souvisí instituty **formální a materiální publicity**. Formální publicita je zásadou, která znamená, že údaje o osobách evidované v rejstříku jsou veřejné a tedy dostupné pro kohokoliv na internetu. Rejstříkový soud nesmí bránit zájemcům v přístupu k těmto údajům nebo požadovat za přístup k nim peníze. Ve 21. století je již běžné, že mnoho informací je na internetu volně k dispozici. Avšak významným rozdílem mezi veřejnými rejstříky a jinými webovými stránkami je bezesporu fakt, že rejstříky jsou vedeny soudy, což může vést u široké veřejnosti ke vzniku důvěry v tyto informace a lidé budou při svém jednání z těchto informací v dobré víře v jejich pravdivost beze strachu vycházet. Proto je nezbytné nastínit i princip tzv. materiální publicity. [45] Ta je upravena v § 8 a § 9 zákona o veřejných rejstřících právnických a fyzických osob. Pokud je určitá skutečnost ve veřejném rejstříku zapsána a třetí osoba právně jedná v souladu s tímto zapsaným údajem, i když již není aktuální a pravdivý, nelze to této osobě vytýkat. [47]

Významným ustanovením z hlediska **agregace dat z veřejných rejstříků** je zejména ustanovení § 25 zákona o veřejných rejstřících PO a FO. To vyjmenovává **skutečnosti do veřejného rejstříku zapisované**.

Půjde o tyto údaje:

- jméno a sídlo nebo adresa místa pobytu,
- předmět činnosti nebo podnikání,
- právní forma právnické osoby a den jejího vzniku a zániku,
- u každé fyzické osoby datum narození, rodné číslo a adresa místa pobytu,
- identifikační číslo osoby, které jí přiděluje rejstříkový soud,
- informace o statutárním orgánu,
- způsob, jakým člen statutárního orgánu jedná za právnickou osobu,
- pokud se zřizuje kontrolní orgán, informace o něm,
- údaje o prokurách,
- další skutečnosti,
- den, kdy byl zápis proveden. [41]

2.1.1 Obchodní rejstřík

Dle § 42 písm. a) zákona o rejstřících právnických a fyzických osob se do obchodního rejstříku zapisují **obchodní společnosti a družstva** podle zákona č. 90/2012 Sb., zákon o obchodních korporacích a družstvech. [38] Podle tohoto zákona rozlišujeme mezi osobními a kapitálovými obchodními společnostmi. Do osobních obchodních společností patří veřejná obchodní společnost a komanditní společnost. Kapitálové obchodní společnosti jsou společnost s ručením omezeným a akciová společnost. Družstva pak zákon člení na bytová a sociální družstva. Dle § 42 písm. b) zákona o rejstřících právnických a fyzických osob se do obchodního rejstříku zapisují **podnikající fyzické osoby**, které mají trvalé bydliště v České republice a o takový zápis požádají. Podle občanského zákoníku je podnikatelem osoba, která samostatně vykonává výdělečnou činnost na vlastní účet a odpovědnost s cílem činit tak soustavně za účelem zisku, a to živnostenským nebo podobným způsobem. Dále se do obchodního rejstříku zapíší osoby, které uvádí § 43 zákona o rejstřících právnických a fyzických osob, které podnikají na území České republiky a o zápis požádají. Dalšími zapisovanými osobami jsou například státní a národní podniky, evropská hospodářská zájmová sdružení, evropské společnosti a jiné. Co se skutečností zapisovaných do obchodního rejstříku týče, je nutné se zaměřit na § 48 zákona o veřejných rejstřících PO a FO. Toto ustanovení doplňuje § 25 stejnojmenného zákona. Do obchodního rejstříku se tak dále zapisuje například výše základního kapitálu obchodních společností či zástavní právo k podílu v obchodní korporaci. [47]

2.1.2 Insolvenční rejstřík

Insolvenční rejstřík [15] je upraven zákonem č. 182/2006 Sb., zákon o úpadku a způsobech jeho řešení (dále jen “insolvenční zákon”) v § 419 a násl. [32] Dle tohoto zákona je insolvenční rejstřík **veřejně dostupným systémem** spravovaným Ministerstvem spravedlnosti ČR. Insolvenční rejstřík obsahuje seznamy insolvenčních správců, dlužníků a insolvenční spisy. Je možné v něm vyhledat dlužníka podle jeho jména/názvu, rodného čísla, identifikačního čísla nebo spisové značky přiřazené k insolvenčnímu řízení. V insolvenčním rejstříku je též možné dohledat všechna insolvenční řízení, ve kterých byl v daném časovém období prohlášen konkurs na majetek dlužníka nebo bylo vydáno jiné stěžejní rozhodnutí. Je obecně doporučeno pro spolehlivé výsledky vyhledávání vyhledávat dlužníka (právnickou osobu) dle identifikačního čísla. Insolvenční spis dlužníka je rozdělen do oddílů: Oddíl A) obsahuje spis od zahájení insolvenčního řízení do rozhodnutí o úpadku a Oddíl B) obsahuje pokračování spisu od rozhodnutí o úpadku až do konce insolvenčního řízení. V oddílu C) jsou uvedeny údaje týkající se incidenčních sporů (incidenční žaloby a rozhodnutí o nich) a v oddílu D) jsou uvedeny údaje jako záznamy o nahlédnutí do spisu či dotazy na stav řízení. Obsahem posledního oddílu P) jsou přihlášky věřitelů. V insolvenčním rejstříku jsou publikovány plné texty dokumentů, případně jejich anonymizované verze. Co se zásadních soudních rozhodnutí týče, uvádí se datum, kdy rozhodnutí nabylo právní moci. Díky insolvenčnímu rejstříku mohou lidé z pohodlí svých domovů nebo kanceláří procházet obsah insolvenčních spisů, jako by fyzicky nahlíželi do spisů insolvenčního soudu. [55]

Insolvenční rejstřík nabízí také webové služby pro automatizovaný přístup a získávání dat. Služby jsou přístupné pomocí rozhraní *SOAP* a podle popisu jsou koncipovány pro poskytnutí maximálního množství informací a zachování vysoké dostupnosti. První ze služeb je *ISIR WS_1* která umožňuje získání všech zaznamenaných událostí napříč řízeními chronologicky od specifikované události. Webová služba předpokládá vlastní rekonstrukci databáze na straně uživatele z důvodu požadavku na vysokou dostupnost. Druhou službou

je ISIR_CUZZK_WS2 umožňující dotázání na konkrétní řízení podle identifikačních údajů dlužníka či spisové značky. Služba však poskytuje pouze stručné informace o řízení a slouží tak primárně k ověření jeho stavu či existence záznamu pro zadaný subjekt. Neposkytuje detailní informace o průběhu řízení ani zveřejněné dokumenty. [52]

2.1.3 Živnostenský rejstřík

Živnostenský rejstřík [40] je informační systém spravovaný Ministerstvem průmyslu a obchodu a provozovaný obecními a krajskými živnostenskými úřady. Obsahuje **údaje o podnikatelích, fyzických i právnických osobách, podle živnostenského zákona**, včetně změn těchto údajů. Jde zejména o jméno, příjmení, identifikační číslo osoby, sídlo, adresu bydliště, předmět podnikání, adresu provozovny či dobu platnosti živnostenského oprávnění. Tyto údaje jsou veřejné, s výjimkou údajů o pokutách udělených živnostenskými úřady a sankčních opatřeních udělených jinými správními orgány v souvislosti s podnikáním, údajů o bydlišti a místě pobytu v České republice, bydlišti mimo území ČR a rodných číslech, které jsou poskytovány pouze podnikatelům, ke kterým se vztahují, správnímu orgánu, který potřebuje tyto údaje pro výkon své činnosti a v případech stanovených zvláštními právními předpisy. Informace z živnostenského rejstříku jsou kromě webové stránky přístupné pomocí veřejného XML rozhraní, skrz které je možné vytvořit stejné dotazy jako pomocí formuláře na webové stránce. Lze tak jednoduše získat data ve strukturované formě v XML formátu. [22]

2.1.4 Katastr nemovitostí

Katastr nemovitostí je jedním z nejdůležitějších veřejně dostupných seznamů. Významnými právními předpisy, které upravují katastr nemovitostí v České republice, jsou zákon č. 256/2013 Sb., zákon o katastru nemovitostí [33] a zákon č. 359/1992 Sb., zákon o zeměměřičských a katastrálních orgánech [35]. Katastr nemovitostí je **veřejný seznam**, který obsahuje údaje o nemovitostech včetně jejich soupisu, popisu, geometrického a polohového určení a zápisu práv k nemovitostem. Tato evidence také zahrnuje vlastnická a jiná věcná práva a další práva stanovená zákonem. Katastr nemovitostí poskytuje **důležité informace o pozemcích a stavbách** a jejich vlastnících a slouží k ochraně práv k nemovitostem, daňovým a poplatkovým účelům, ochraně životního prostředí, rozvoji území, oceňování nemovitostí, vědeckým, hospodářským a statistickým účelům a tvorbě dalších informačních systémů. Pro účely této diplomové práce je důležitá informace obsažená v odst. 2 písm. b) zákona o katastru nemovitostí, která předvídá a umožňuje použití údajů k tvorbě dalších informačních systémů, pokud neslouží jiným účelům než jsou uvedeny v odst. 1 tohoto ustanovení. [44]

Aplikace *Nahlížení do katastru nemovitostí* [19] poskytuje možnost procházet údaje o parcelách, budovách, jednotkách a právech k nemovitostem, které jsou evidovány v katastru nemovitostí v České republice. Přístup do katastru nemovitostí je **bezplatný** a nevyžaduje registraci, avšak anonymní nahlížení do katastru nemovitostí bylo omezeno. Rozhodl tak ústřední orgán státní správy v oblasti evidence nemovitostí, Český úřad zeměměřičský a katastrální. Aplikace je určena **pouze pro interaktivní práci uživatelů**, automatizované získávání nebo vytěžování dat není dovoleno. Existuje placená služba *Dálkový přístup k údajům katastru nemovitostí České republiky* [10], která nabízí webové rozhraní pro možnost automatizovaného přístupu. Poplatky jsou účtovány za jednotlivě provedené operace od 10 Kč do 100 Kč. Vzhledem k této povaze není propojení s katastrem nemovitostí dále uvažováno v rámci této práce.

2.1.5 Administrativní registr ekonomických subjektů

Administrativní registr ekonomických subjektů, známý pod zkratkou ARES, je **informační systém**, který zpřístupňuje veřejné údaje o ekonomických subjektech z různých zdrojů veřejné správy.

Tyto zdroje zahrnují například:

- veřejný rejstřík, který vedou rejstříkové soudy,
- živnostenský rejstřík, vedený Ministerstvem průmyslu a obchodu ČR,
- registr ekonomických subjektů, vedený ČSÚ,
- registr církví a náboženských společností, vedený Ministerstvem kultury,
- národní registr poskytovatelů zdravotních služeb, vedený Ústavem zdravotnických informací a statistiky ČR,
- evidenci zemědělského podnikatele, vedenou Ministerstvem zemědělství ČR,
- seznam politických stran a hnutí, vedený Ministerstvem vnitra ČR,
- registr plátců daně z přidané hodnoty,
- registr plátců DPH, vedený Finanční správou ČR,
- centrální evidenci úpadců, vedenou Ministerstvem spravedlnosti ČR,
- insolvenční rejstřík, vedený Ministerstvem spravedlnosti ČR.

Tyto zdroje jsou doplněny o údaje z dalších zdrojů, jako je například Registr dotací z rozpočtu nebo Registr územní identifikace, adres a nemovitostí vedený Českým úřadem zeměměřickým a katastrálním. ARES je pravidelně aktualizován, aby zajistil aktuálnost veškerých informací. Je však třeba mít na paměti, že aktuálnost informací závisí na možnostech informačních systémů jednotlivých zdrojových registrů. Samotný ARES upozorňuje na fakt, že *“informace zpřístupněné systémem ARES mají pouze informativní charakter a nemohou být použity jako průkazný podklad pro soudní řízení ani nemůže být požadována náhrada škody, která by vznikla jejich využitím.”* [3]

ARES je důležitým nástrojem pro všechny, kteří potřebují získat informace o ekonomických subjektech, a může být použit k mnoha různým účelům, například k ověření údajů o subjektu při obchodních jednáních, k přípravě výběrového řízení na zakázku či k získání informací pro účely daňového řízení. Systém ARES navíc nabízí sérii volně přístupných webových služeb, které umožňují získání dat ve strukturované podobě ve formátu XML.

2.2 Informace o soudních řízeních

Ministerstvo spravedlnosti České republiky je zodpovědné za fungování justice v zemi a zajišťuje, aby byla zachována její nezávislost a nestrannost. K tomu používá různé nástroje, jako je například aplikace *ePodatelna*. [4] Ta slouží k **elektronické komunikaci se soudem** a od 1. října 2007 umožňuje odesílání podání v elektronické formě. Dalším důležitým nástrojem pro elektronické soudnictví je aplikace *infoSoud*. [14] Aby bylo možné získat **informace o stavu projednávání případu**, je nutné nejprve zadat příslušnou spisovou

značku. Cílem Ministerstva spravedlnosti je zajistit co nejvyšší otevřenost justice a umožnit veřejnosti snadný přístup k informacím o soudních řízeních. Aplikace *infoSoud* a další nástroje elektronického soudnictví, například *infoJednání* [13], pomáhají účastníkům řízení a jejich právním zástupcům snadno získat přehled o probíhajících soudních řízeních a sledovat postup v řešení jejich případů.

2.2.1 InfoSoud

Od roku 2008 je široké veřejnosti, zejména pak účastníkům řízení, k dispozici služba zvaná *infoSoud* [14] od Ministerstva spravedlnosti, která **umožňuje zobrazovat informace o stavu konkrétního soudního řízení** na libovolném soudu v České republice. Účastníci řízení mohou sledovat průběh řízení v dané věci a sledovat přehled událostí, které nastaly v průběhu řízení, aniž by museli navštěvovat soudy osobně.

Pro začátek je třeba vybrat soud, u kterého řízení probíhá. Možnosti, mezi kterými lze volit jsou: Nejvyšší soud, vrchní soudy, krajské soudy nebo okresní (obvodní soudy). V případě výběru Nejvyššího soudu, je možné přejít rovnou k vyplnění spisové značky. V případě výběru krajského či okresního soudu je třeba volit z nabízených seznamů a následně je třeba vyplnit spisovou značku stejně jako v případě předchozím. Pro snazší hledání okresního soudu je možné vybrat nejprve krajský soud, čímž se vyfiltruje nabídka okresních osudů. Pokud je podle zadaných údajů nalezeno řízení, je zobrazen přehled nastalých událostí, u kterých je možné zobrazit detail. Jedná se např. o události nařízení či zrušení jednání, vydání rozhodnutí, skončení věci a další.



Informace o řízení

Výběr soudu

Typ soudu: vrchní/krajský/okresní soud

Krajský/Vrchní soud: -- Celá ČR --

Okresní soud: []

Spisová značka: [] [] [] / [] []

Obrázek 2.1: Snímek obrazovky vyhledávacího formuláře na stránce infoSoudu.

2.2.2 InfoJednání

Webová služba nazvaná *infoJednání* [13], zpřístupněná veřejnosti Ministerstvem spravedlnosti, si klade za cíl **informovat zejména o časech, kdy probíhá konkrétní jednání**. Na tuto informaci se subjekty jednotlivých soudů tázaly nejčastěji, a proto tato služba výrazně usnadnila onen proces získávání informací. Služba *infoJednání* dovoluje získat následující data:

- číslo jednacích síní,
- čas a datum jednání,

- spisová značka,
- jméno soudce, který jednání řídí,
- předmět jednání,
- účastníky,
- informace o tom, zda je jednání veřejné či nikoliv.

Zpřístupněny mají být informace o všech nařízených jednáních ze soudů v České republice. Aby subjekt získal kýžené informace, je třeba vyplnit formulář. Důležitou informací pro veřejnost je skutečnost, že webová služba *infoJednání* je schopna zobrazit nařízení jednání pouze na následujících 30 dnů.

2.3 Procesy v právní kanceláři

V této části jsou popsány a analyzovány nejdůležitější procesy v běžné advokátní kanceláři a jejich průběh. Důraz je kladen na procesy, které souvisí s informačními technologiemi nebo jimi mohou být podpořeny. Hladký průběh procesů v každé advokátní kanceláři je nezbytný pro její efektivní fungování. Je to však relevantní i pro jednotlivě působící advokáty, pro které jsou mnohé procesy podobné. Dostatečné porozumění procesům běžné advokátní kanceláře bude zásadní pro sestavení specifikací a vytvoření dobrého návrhu požadovaného systému. Následující popis vychází z vlastního pozorování v konkrétní kanceláři a z rozhovorů vedených s několika zaměstnanci tří odlišných advokátních kanceláří. Mimo to bylo čerpáno z monografie Jaroslava Svejkovského, který se profesionálně zabývá advokátním právem. [61]

2.3.1 Získání nového klienta

Získávání nových klientů je proces, kterým musí pravidelně procházet většina běžných advokátních kanceláří i samostatně působících advokátů. Výjimku tvoří některé specializované advokátní kanceláře, které již nové klienty neshánějí a věnují se dlouhodobé spolupráci se současnými klienty, kterým výhradně poskytují své služby. Jedná se zpravidla o větší advokátní kanceláře, jejichž klienty jsou velké korporace, které potřebují neustálou právní podporu. Vyvíjený systém bude orientovaný spíše na **samostatné advokáty či menší advokátní kanceláře**, pro něž je stále získávání nových klientů zásadní pro udržení finančního příjmu a další rozvoj.

Pro získání klientů je zpravidla potřeba určitým způsobem **zviditelnit a prezentovat nabízené právní služby** a informovat tak potenciální klienty o dostupných možnostech řešení jejich problémů. K tomu může být využito několika kanálů.

Mezi tyto kanály patří například:

1. vlastní kontakty a osobní vztahy,
2. webové stránky,
3. sociální média,
4. placená propagace služeb pomocí reklamy na internetu, v místním tisku nebo prostřednictvím dalších mediálních kanálů,

5. reference od ostatních klientů,
6. přímé nabízení služeb vytipovaným firemním zákazníkům skrze email, telefonicky nebo na osobní schůzce.

Tyto kanály mohou představovat první fázi v procesu získávání nového klienta. Jednou z nejúčinnějších metod získávání nových klientů je právě kombinace oněch kanálů. Ta zajistí dosažení co nejširšího okruhu potenciálních klientů a zvýší povědomí o právní kanceláři.

Následujícím krokem v procesu získávání klienta je **první osobní kontakt**. Typicky je navázán ze strany nového potenciálního klienta, který advokáta či kancelář kontaktuje se svým specifickým požadavkem.

Nejčastěji je tak učiněno prostřednictvím jednoho z následujících způsobů:

1. telefonát,
2. osobní návštěva,
3. email,
4. formulář na webových stránkách či jiná zpráva.

Požadavek klienta může spadat do několika kategorií. Může jít o žádost o běžnou právní radu či sepsání smlouvy, pomoc s řešením již existujícího sporu, nebo klient může být podezřelý z trestného činu a advokáta si buď sám vybral nebo mu mohl být i přidělen. Na základě prvního kontaktu je buďto:

- a) Dohodnuta spolupráce s klientem s přibližným stanovením odměny, což je typické pro běžné a menší požadavky, jako je například sestavení kupní smlouvy, nebo k dohodě o spolupráci nedojde.
- b) Sjednání delší osobní schůzky, případně telefonického nebo online setkání, během kterého budou probrány podrobnější okolnosti a podmínky budoucí spolupráce. Na takové schůzce má advokát možnost se seznámit s požadavky klienta a s konkrétním případem, který má řešit. Advokát klientovi představí možné cesty řešení, navrhne průběh spolupráce a seznámí jej s předpokládanými náklady. Po ukončení schůzky je rozhodnuto o tom, zda dojde k uzavření dohody o spolupráci, či nikoliv.

2.3.2 Založení a vedení spisu

Advokátní kanceláře si pro každého svého klienta vedou spisy, které obsahují stěžejní informace o klientovi a podrobné informace o řešených případech, související dokumenty či komunikaci se zainteresovanými stranami a orgány. Povinnost advokátů vést spis je mimo jiné stanovena v **§ 25 odst. 1 zákona o advokacii**. [36] V literatuře bývá uváděno, že spisy jsou vedeny zejména ve prospěch klientů, ovšem dle Svejkovského slouží vedení spisu **primárně k ochraně advokáta**. Dokumentace určuje jeho důkazní pozici při prokazování řádného a včasného poskytování právních služeb a jejich řádného vyúčtování. Stejně tak dokládá soulad spisu se zákonnými a stavovskými povinnostmi. Dokumentaci musí advokát vést takovým způsobem, aby z ní byl zřejmý jeho postup v rámci poskytování právních služeb a zejména oprávněnost vyúčtované odměny. Správa a vedení spisů jsou tak centrálním procesem každého advokáta a advokátní kanceláře. Přináší přehled o všech aktivitách a závazcích vůči klientům a jsou tak nezbytné pro efektivní vykonávání jejich činností a poskytování kvalitních právních služeb. Ve složce každého klienta může být uchováváno několik

případů, které s advokátní kanceláří řeší nebo řešil v minulosti. V rámci případu jsou vedeny podrobnosti případu, smlouvy o uzavření spolupráce mezi klientem a advokátní kanceláří, informace o stranách vystupujících v rámci případu, historie korespondence a veškeré další související dokumenty, které se v průběhu řešení případu objeví a jsou postupně přidávány. Může se jednat též o nové verze dokumentů již ve spisu přiložených. To, jakým způsobem má být advokátní dokumentace vedena, není zákonem přesně stanoveno. Advokát vede tzv. **klientský spis**, obsahující dokumenty týkající se konkrétního klienta či konkrétní právní záležitosti. Je nutné poznamenat, že poskytování právních služeb je **činností soukromou** a zejména je nutno klást důraz na účel, pro který je spis veden. [61]

Obsah advokátního spisu:

Advokátní spis obsahuje smluvní ujednání s klientem, plnou moc či doklad o jejím udělení, záznam o klientem schváleném postupu ve věci a pokyny klienta, které mohou být písemné, e-mailové nebo zaznamenané ústní pokyny. Plná moc advokátovi umožní vykonávání právních úkonů jeho jménem a zastupování klienta před soudy, úřady či jinými institucemi. Dále by měl spis zahrnovat komunikaci mezi advokátem a klientem, podklady pro poskytování právních služeb od klienta nebo jiných osob, listiny jako je smluvní dokumentace a podání, a doklady o odeslání, jako jsou dodejky či doručky. V případě, že advokát vede tzv. knihu doručené/odeslané pošty, není třeba založení zmiňovaných dodejek a doruček a postačí na dokumenty datum doručení/odeslání poznamenat. [61] Není samozřejmě vyloučeno, aby spis obsahoval rešerše relevantní judikatury.

Další dokumentace:

Existují specifické situace, ve kterých advokát musí vést tzv. **další dokumentaci**. Ta nemusí být nutně součástí klientského spisu. Taková dokumentace slouží primárně stejně jako advokátní spis k **prokázání splnění zákonných a stavovských povinností**. Sekundárně může mít i jiné účely, třeba dokazování pravosti podpisu. V rámci provádění autorizovaných konverzí dle **§ 26 zákona o elektronických úkonech** [34] musí vést advokát zvláštní evidenci obsahující údaje tímto zákonem předepsané. Jedná se například o pořadové číslo, pod kterým je konverze vedena v evidenci provedených konverzí či datum provedení konverze. Každý advokát by měl dále evidovat plnění daňových, pojišťovacích, pracovněprávních a dalších evidenčních povinností. Advokátní kancelář lze bezesporu vnímat též jako obchodní a účetní jednotku, proto by advokát měl evidovat i nákladovou stránku zastupování, depozita mimo úschovu peněz a majetek kanceláře. [61]

Způsoby, kterými mohou být spisy vedeny:

a) Listinné vedení spisu

První možností je klasické vedení spisu a veškeré dokumentaci pouze ve **fyzické papírové podobě**. V tomto případě se jedná o běžný šanon, který obsahuje veškeré zmíněné informace v podobě papírových dokumentů, které mohou být organizovány do jednotlivých podsložek. Veškerá korespondence či dokumentace, která je případně obdržena elektronickou formou je vytisknuta a přiložena do složky. Veškeré další dokumenty jsou pravidelně přikládány do složky, kterou je nutné uchovávat na bezpečném a zároveň přístupném místě v prostorách advokátní kanceláře. Papírové vedení spisu se ovšem pro mnohé advokáty stává minulostí a v dnešní době je často debatována otázka elektronizace papírových spisů. Je však nutné pamatovat na konzervativnost advokátního stavu.

b) Kombinace elektronického a listinného spisu

Druhou možností je **duplicitní vedení ve fyzické i digitální podobě**. V tomto případě jsou veškeré spisy a dokumentace vedeny jak ve fyzické tak digitální podobě v poměru blízkém 1:1. Při tomto vedení jsou veškeré fyzické dokumenty skenovány a organizovány v digitální podobě a zároveň zakládány do fyzických složek a uchovávány. Obdobně veškeré dokumenty a korespondence obdržené v elektronické podobě jsou evidovány elektronicky, ale zároveň jsou vytisknuty a přiloženy do fyzických složek jako v předchozím případě. Toto duplicitní vedení může advokátní kanceláři přinést **větší bezpečnost a flexibilní přístup**. Na druhou stranu duplicitní vedení představuje **administrativní zátěž** s udržováním aktuálních verzí fyzické a digitální složky v souladu pomocí skenování a tisknutí dokumentů a přináší riziko vzniku ne-konzistence, a zmatku. Co se týče digitálního vedení, to je možné vykonávat také více způsoby více přiblíženým v následující části. Na základě preferencí advokáta a klienta se zpravidla primárně pracuje s jedním typem uchování a druhý typ slouží spíše jako záloha. Preferencí se stává digitální podoba.

c) Primát elektronického spisu nad listinným

Další možností je vedení spisu výhradně v primárně digitální podobě. Dokumenty, které není třeba evidovat ve fyzické podobě, jsou vedeny pouze digitálně s případnou digitální zálohou. Dokumenty obdržené ve fyzické podobě jsou skenovány a vedeny elektronicky. K tomuto je často využíváno tzv. **autorizované konverze dokumentů**, která je upravena zákonem č. 300/2008 Sb., o elektronických úkonech a autorizované konverzi dokumentů [34]. Tato konverze zajišťuje možnost ověření pravosti dokumentu a jeho identitu s fyzickou verzí. Tento elektronický dokument poté plnohodnotně zastupuje fyzický dokument. V advokátním stavu lze pozorovat trend postupné digitalizace a přesunu od papírového vedení spisu k využívání řady systémů, které podporují elektronické vedení klientské dokumentace. Těm je věnována pozornost v následujících odstavcích.

Skartace spisu:

Advokát může spis po uplynutí lhůty pro uchovávání **skartovat, tedy zničit**. Způsob skartace není přesně stanoven. Zničení spisu může provést advokát nebo jiný subjekt na smluvním základě tak, aby bylo zajištěno trvalé zničení obsažených údajů a znemožnění jejich obnovení. Při skartaci je nutné brát ohled na povinnost mlčenlivosti advokáta. [61]

2.3.3 Digitální vedení spisu

Advokátní kanceláře se stejně jako ostatní firmy posouvají směrem k digitalizaci. Mezi jednotlivými kancelářemi se ovšem vyskytují velké rozdíly v tom jak k digitalizaci přistupují a do jaké míry dostupné technologické prostředky využívají. Je důležité poznamenat, že zásady vedení digitálního spisu jsou totožné jako u jeho listinné formy, neexistuje tedy žádná stanovená norma, jak by měl digitální spis vypadat. Svejkovský však upozorňuje na rizika ukládání dokumentace ve virtuální podobě. Dle jeho názoru *"jsou-li dokumenty ukládány na cloudech nebo jiných virtuálních úložištích, musí jít o úložiště, jež jsou schopna zajistit zákonem požadovaný standard ochrany povinnosti mlčenlivosti z hlediska bezpečnosti."* [61] Pokud se kanceláře rozhodnou pro vedení spisů v elektronické podobě, existují prakticky dvě možnosti, jak toho mohou dosáhnout.

První více přímočarou možností je **uchovávání veškerých spisů a dokumentů v běžných složkách počítače**. Souborový systém může být sdílen v místní síti tak, aby k sdíleným souborům mělo přístup několik zaměstnanců zároveň. Veškeré elektronické doku-

menty jsou tak ukládány lokálně na discích počítačů advokátní kanceláře a jsou organizované do vlastní hierarchie složek. Tento přístup se zdá být jako převládající pro digitální vedení spisů, jelikož byl využíván čtyřmi z pěti dotázaných advokátních kancelářích.

Druhou možností je využívání **specializovaného softwaru pro právní kanceláře**. Ty typicky nabízí sofistikovanější funkce pro správu klientů, jejich spisů a souvisejících dokumentů. Umožňují více možností zobrazení a organizace těchto dat. Kromě nástrojů pro správu klientů a organizaci spisů nabízejí mnoho dalších funkcí, ať už v základní verzi, či za příplatek v podobě rozšiřujících modulů. Systémy se liší v nabízené funkcionalitě a možnostech přístupu k datům. Existujících systémů existuje několik a jejich možnosti jsou popsány v následující kapitole “Právní informační systémy”. Obdobný systém bude vytvářen i v rámci této práce.

Obě možnosti mají své výhody a nevýhody. K výhodám první možnosti, uchovávání veškerých spisů a dokumentů v běžných složkách počítače, patří jistě to, že je levnější, protože není potřeba platit licence a poplatky za využívání externího softwaru. Může být pro uživatele jednodušší, jelikož jsou na práci se souborovým systémem zvyklí z běžné práce na PC. Může však být méně efektivní pro vyhledávání konkrétních spisů a dokumentů při větším množství dokumentů, obzvlášť pokud jsou hůře organizované. Je zde také riziko ztráty dat při poškození nebo ztrátě počítače nebo disků. K výhodám druhé možnosti, tedy využívání specializovaného softwaru pro právní kanceláře, patří efektivnější vyhledávání a organizace dat a širší funkcionalita pro správu klientů a spisů. Přínosná je možnost jednoduchého rozšíření funkcionality pomocí dalších modulů a zjednodušení některých činností advokátní kanceláře. Často zahrnuté automatické zálohování může zamezit poškození nebo ztrátě dat. Nevýhody této možnosti jsou vyšší náklady, nutnost zaškolení uživatelů a vznik jisté závislosti na poskytovateli konkrétního softwaru.

Právní kanceláře mohou samozřejmě využívat i různých hybridních přístupů, kdy například pro nějaké účely a případy využívají specializovaný software a pro jiné se zase přiklání k lokálnímu využití možností souborového systému.

2.3.4 Tvorba smluv pro klienta

Tvorba smluv pro klienta v běžné české právní kanceláři zahrnuje několik kroků. Pro potřeby tvorby systému je zde přiblížen přehled procesu tvorby smluv. Nejprve je nutné identifikovat **potřeby konkrétního klienta**, tedy výsledků, kterých chce klient smlouvou dosáhnout. Stěžejní částí v práci právníků je **studium právních předpisů a judikatury soudů**. Poté, co stanoví právník předmět smlouvy, zkoumá platné právní předpisy, které se předmětu smlouvy týkají, aby zajistil její soulad s nimi. Dále právník **získává data z veřejných rejstříků a seznamů**, jimž byl prostor věnován výše. Dále vytvoří **návrh smlouvy**, jehož náležitosti závisí na konkrétním smluvním typu. Klientovi tento návrh smlouvy následně prezentuje, ten případně může navrhnout změny či požádat právníka o doplnění. Po případných revizích dochází k **finalizaci smlouvy** a přejde se k jejímu podpisu. Jednou ze zásadních náležitostí, kterou musí nezbytně obsahovat každá smlouva je **řádná identifikace stran**. Zaměstnanci advokátní kanceláře proto musí disponovat aktuálními údaji o stranách smlouvy, jako jsou jména, adresy či identifikační čísla. Takové údaje vyhledávají a jejich správnost ověřují zaměstnanci advokátních kancelářích především v již zmiňovaných veřejných rejstřících. Předmětem kupních, darovacích a dalších smluv může být často nemovitá věc. Ta musí být detailně popsána, aby bylo jasné, o kterou nemovitost se jedná. Přístup k těmto informacím usnadňuje již zmiňovaný katastr nemovitostí. Jelikož se údaje v rejstřících mohou měnit, je třeba, aby byli právníci bdělí a údaje pravidelně kontrolovali,

tedy nespolehali na informace, které od klientů získali například při prvním pohovoru. To může být ovšem časově náročné a právě systém pro agregaci dat z veřejných databází si klade za cíl celý tento proces výrazně zjednodušit a pravidelně získávaná data aktualizovat.

2.3.5 Komunikace s úřady a soudy

V rámci práce právníka může komunikace s úřady a soudy zahrnovat širokou škálu činností, jako je zejména zastupování klienta před nimi na základě plné moci, podávání dokumentů, či vyžádání informací na základě zákona č. 106/1999 Sb., zákon o svobodném přístupu k informacím [31], dle kterého mají povinné subjekty povinnost poskytovat informace vztahující se k jejich působnost. Zákon mezi ně řadí zejména státní orgány, územní samosprávné celky a jejich orgány a veřejné instituce. Může se jednat například o finanční úřady, stavební úřady či katastrální úřady. Na webové stránce www.statnisprava.cz [39] jsou zpřehledněny Úřady v České republice dle krajů a lze se tak dostat k údajům o nich, jako například k webovým stránkám, e-mailům, adresám, či telefonním číslům. Právník může podávat žádosti, návrhy, stížnosti nebo jiné dokumenty jménem klienta na příslušných úradech nebo soudech. Mezi zdroje, které mohou být pro právníky užitečné v rámci komunikace s úřady a soudy patří zejména *portál Justice.cz* [21], jenž je oficiálním a specializovaným informačním portálem veřejné správy, který sdružuje nejen informace ze soudnictví v České republice. Na tomto portále jsou vyhledatelné informace o soudech, konkrétně je zde dostupný jejich přehled a seznam soudců. Dále portál obsahuje informace o státních zastupitelstvích.

2.3.6 Řešerše pro klienty

Při provádění rešerše judikatury pro klienty lze pracovat s následujícími zdroji informací, které patří mezi nejčastěji využívané:

- Webová stránka Nejvyššího soudu České republiky [18], kde je zpřístupněn vyhledávač rozhodnutí a stanovisek Nejvyššího soudu. Ten přináší jak odborníkům, tak laikům pohodlný přístup k rozhodnutím a stanoviskům Nejvyššího soudu. Systém umožňuje snadné vyhledávání rozhodnutí podle spisových značek, stejně jako hledání rozhodnutí týkajících se konkrétních právních otázek, prostřednictvím fulltextového vyhledávání nebo dalších parametrů v rozšířeném vyhledávacím formuláři. Rozhodnutí jsou doplněna o dodatečné informace (právní věty, použité právní předpisy, klíčová slova), která lze také použít pro vyhledávání. Co se rozhodnutí vrchních a krajských soudů na internetu týče, Ministerstvo spravedlnosti usilovně pracuje spolu s Nejvyšším soudem na uvedení vyhledávání v judikatuře vrchních a krajských soudů pro veřejnost. Nyní je tato databáze ve zkušebním provozu a obsahuje vybraná rozhodnutí vrchních, krajských a v některých případech i okresních soudů. [11]
- Databáze Evidence soudních rozhodnutí Nejvyššího správního soudu. [27]
- Databáze Evidence soudních rozhodnutí Ústavního soudu ČR. [8]
- Databáze vybraných rozhodnutí Evropského soudu pro lidská práva. [9]
- ASPI (Autorský systém právních informací), který je placeným systémem právních informací a obsahuje databáze soudních rozhodnutí, právních předpisů, odborné literatury a časopisů. [30]

- Codexis je právní informační systém, který nabízí přístup k právním předpisům, soudním rozhodnutím, odborné literatuře a časopisům. Jedná se o placenou službu. [7]
- Beck Online je též placeným právním informačním systémem. Je známý pro svou komplexnost a širokou škálu zdrojů, které pokrývají různé oblasti práva. [6]

Kapitola 3

Informační systémy a existující řešení

V této kapitole jsou shrnuty nejdůležitější poznatky z nastudované teorie k obecné tvorbě informačních systémů a návrhu uživatelských rozhraní. Následně je proveden průzkum existujících právních informačních systémů a analýza jejich funkcionality. Na závěr jsou přiblíženy vybrané implementační technologie. Výstupem této kapitoly je solidní teoretický základ pro vytvoření kvalitního návrhu systému a jeho následnou implementaci.

3.1 Tvorba informačních systémů

V této podkapitole je přiblížena problematika informačních systémů a jejich tvorby. Je pospáno co jsou informační systémy a jakým účelům slouží včetně toho jaké byly motivace pro jejich vznik. Jsou zde uvedeny často používané architektury informačních systémů a jejich vlastnosti. Porozumění této problematice bude nápomocné při analýze konkrétních existujících systémů v následující podkapitole a nabyté znalosti budou též sloužit jako další z podkladů při vlastním návrhu vytvářeného systému v kapitole třetí.

3.1.1 Definice, rozvoj a rozdělení

Na informační systémy existuje několik možných pohledů a je možné je definovat různými způsoby. Stejně tak existuje několik typů informačních systémů a jsou využívány různé metodologie pro jejich tvorbu. Podle jedné z používaných definic je informační systém takový systém, který **umožňuje účelné uspořádání sběru, uchování, zpracování a poskytování informací**. Informační systémy začaly vznikat s rozvojem prostředků v oblasti informačních technologií. S rozvojem osobních počítačů a později internetu začala vznikat potřeba efektivně nakládat s informacemi v elektronické podobě. Bylo důležité mít možnost je efektivně organizovat, přistupovat k nim, sdílet je napříč organizací a získávat z nich užitečné znalosti. Veškeré tyto požadavky vedly ke vzniku a rozvoji různých informačních systémů, které pomáhaly řešit právě tyto a další problémy. V dnešní době může být požadavků na systémy více a s postupem technologií jsme schopni dělat informační systémy komplexnější, s širším rozsahem funkcionality a chytřejší pro práci s daty. Základní motivace pro jejich využívání se však zásadně nezměnily.[51]

Informační systémy je možné rozdělit z několika různých pohledů, jsou zde proto vybrány jen ty nejčastější způsoby rozdělení. Jedním ze základních rozdělení může být rozdělení na podnikové informační systémy a veřejné informační systémy. [42]

Podnikové informační systémy jsou využívány v soukromých organizacích a společnostech a typicky jsou využívány pro zefektivnění a lepší organizaci různých podnikových procesů. Jejich účelem bývá zjednodušit práci zaměstnancům i vedení firmy. Zároveň poskytují přehled o podnikových aktivitách a umožňují jednoduchý přístup k různým datům, které organizace uchovává.

K některým častým příkladům využití může patřit například:

- správa a uchovávání dat o zákaznících, dodavatelích a produktech,
- řízení skladových zásob a nákupních objednávek,
- zpracování faktur a vyúčtování,
- plánování a řízení výroby,
- správa a plánování pracovních míst a lidských zdrojů,
- řízení projektů a zadávání úkolů,
- sledování a vyhodnocování finančních výsledků podniku.

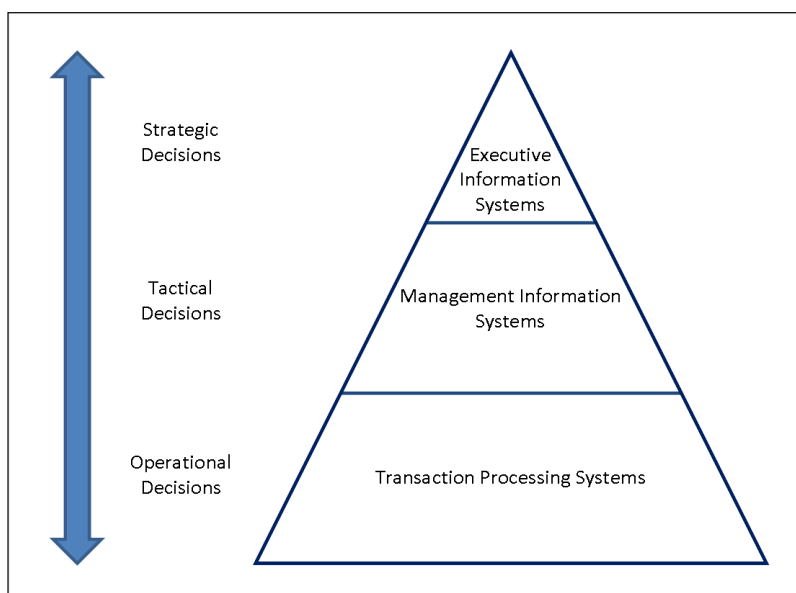
Veřejné informační systémy se liší od podnikových informačních systémů tím, že jsou dostupné pro širokou veřejnost a nejen pro zaměstnance jedné organizace a jejich přístup tak není omezen pouze na určitou skupinu oprávněných osob. Tyto systémy poskytují přístup k různým informacím, které jsou uchovávány a nabízeny institucemi, jako jsou knihovny, muzea nebo státní a další veřejné instituce. Systémy jsou tak často financované z veřejného rozpočtu a mohou nabízet různou funkcionalitu. Například knihovny mohou prostřednictvím takového systému poskytovat informace o dostupných knihách, výpůjčních podmínkách, nabízených službách a zájemci si skrz takový systém mohou třeba zarezervovat vypůjčení vybrané knihy či prodloužit existující výpůjčku. [42] Rejstříky a online zdroje popisované na začátku této kapitoly by bylo také možné zařadit právě do této kategorie.

Informační systémy jsou dále často děleny **podle způsobů využití**. Jedná se pak o více specializované informační systémy sloužící konkrétnímu účelu a nabízející pokročilejší funkcionalitu ve specifickém směru. Mezi nejčastější typy informačních systémů podle využití patří například následující:

- ERP (Enterprise Resource Planning) je typ informačního systému, který se využívá pro plánování a řízení podnikových zdrojů, jako jsou například zásoby, výroba, financování, lidské zdroje a prodeje. Bývá primárně využíván vedením společnosti a je všeobecně považován za nejčastější typ informačního systému.
- MIS (Management Information System) je informační systém určený pro poskytování aktuálních přehledů o chodu podniku pro vedoucí pracovníky a managery a může jim pomáhat s plánováním a rozdělováním práce v týmech.
- SCM (Supply Chain Management) je informační systém zaměřený na řízení dodavatelského řetězce a efektivní a hladký chod celého procesu od nákupu surovin až po prodej hotového výrobku. Hodí se pro velké výrobní podniky.
- HRM (Human Resource Management) je informační systém sloužící k řízení lidských zdrojů v organizaci a umožňuje lepší správu a plánování pracovních míst.

- MES (Manufacturing Execution System) je informační systém určený pro řízení výrobních procesů. Pomáhá zajistit, aby výroba probíhala efektivně a v souladu s plánem.
- CRM (Customer Relationship Management) je informační systém sloužící k řízení vztahů se zákazníky. Umožňuje lepší správu kontaktů a pomáhá se zlepšením komunikace s nimi. [56]

Dělení systémů je též možné podle toho na jaké úrovni v organizaci jsou primárně využívány a jaké typy rozhodování tak podporují. Mluví se pak o “*Exekutivních informačních systémech*” využívaných nejvyšším vedením firmy pro podporu strategického rozhodování, “*Manažerských informačních systémech*” ve střední úrovni firmy pro podporu rozhodování managerů ohledně jednotlivých týmů a taktických rozhodnutí a na nejnižší úrovni “*Systémy transakčního zpracování*”, které jsou zaměřené více na běžné “rutinní úkony”, jako zaznamenávání dat, zpracování objednávek apod. [53]



Obrázek 3.1: Pyramidový model pro pohled na informační systémy rozdělené podle místa využití v organizaci. Převzato z: http://www.chris-kimble.com/Courses/World_Med_MBA/Types-of-Information-System.html

3.1.2 Architektury

Existuje mnoho různých architektur informačních systémů, které se liší podle toho, z jakých hlavních částí se systém skládá a jakým způsobem spolu tyto části komunikují. Mezi nejčastěji používané architektury patří například:

Klient-server

Klient-server je architektura, která se skládá ze dvou hlavních částí: klienta a serveru. Klient je počítač nebo zařízení, na kterém jsou využívány služby informačního systému. Je zodpovědný za zobrazování informací uživateli, zpracovává akce vykonané uživatelem a komunikuje se serverem přes síť. Tomu posílá požadavky na zpracování dat nebo získání

informací. Server je počítač, který hostuje celý systém. Je tak zodpovědný za většinu zpracování a ukládání dat. Reaguje na požadavky ze strany klienta a poskytuje mu odpovědi ve formě vyžádaných dat. [46]

Třívrstvá architektura

Třívrstvá architektura je architektura, která vychází z architektury klient-server ale nabízí k ní další rozšíření a tím ji vylepšuje. Skládá se ze tří hlavních vrstev, kterými jsou:

1. Prezentační vrstva: zodpovídá za zobrazování informací uživatelům a za komunikaci mezi uživatelským rozhraním a zbývajícími dvěma vrstvami.
2. Aplikační vrstva: zodpovídá za logiku aplikace a za zpracování dat. Data získává od datové vrstvy a ve zpracované formě je vrací na stranu klienta
3. Datová vrstva: zodpovídá za ukládání a získávání dat z databáze nebo jiného úložiště. Je zastupována typicky databází nebo jiným úložištěm dat.

Tato architektura je v dnešní době využívána většinou systémů a aplikací pracujících s daty. Může se jednat informační systémy ale i webové aplikace a portály. Výhodou architektury je pružnější rozdělení výkonu mezi zařízením na straně uživatele a server. Klient je často nazýván jako *thin client* (*tenký klient*) symbolizující opravdu minimální požadavky na jeho výkon, jelikož většina zpracování se děje na straně serveru. Prezentační vrstva tak může běžet i na velmi levných zařízeních. Mezi další výhody patří například možnost efektivnějšího vývoje, jelikož týmy mohou pracovat současně na jednotlivých vrstvách a také dobrá škálovatelnost, jelikož každá vrstva může být rozšířena samostatně bez závislosti na ostatních vrstvách. [29]

Monolitická architektura

Monolitická architektura je způsob organizace softwarového systému, ve kterém jsou všechny komponenty aplikace, jako je prezentační vrstva, aplikační vrstva a datová vrstva, integrovány do jedné kódbáze a běží jako jeden samostatný celek.

Mezi výhody monolitické architektury patří jednodušší vývoj, testování a nasazení, protože celá aplikace je součástí jednoho celku. Správu zdrojového kódu je jednodušší, protože je veškerý soustředěn na jednom místě a stejně tak celá aplikace může být zveřejněna jako jeden celek.

Nevýhodou je obtížnější škálovatelnost - jelikož celý systém musí být škálován najednou, může to být více nákladné. Dále u této architektury může také pravděpodobněji nastat vyšší závislost a provázání mezi jednotlivými komponentami, což může vést k složitějšímu provádění změn. Aktualizace jedné části systému může snadno ovlivnit všechny ostatní části a spolupráce více týmů může být obtížnější na koordinaci. [50]

Microservices architektura

S využitím této architektury je aplikace rozdělena do menších, samostatně nasaditelných služeb, které spolupracují, ale každá má svou vlastní oblast zodpovědnosti, datové úložiště a komunikační rozhraní. Jednotlivé služby spolu komunikují (typicky pomocí API) pro dosažení požadované funkcionality.

Výhody mikroservices architektury zahrnují lepší škálovatelnost, jelikož každá služba může být rozšiřována, testována a nasazena nezávisle na ostatních. S tím je i jednodušší rozdělení práce mezi týmy a každý může pracovat nezávisle na vybrané službě.

Nevýhodou je zvýšení komplexity správy celého systému vzhledem k nutnosti správy a monitorování několika samostatných částí. Může přinést také větší zátěž pro mezitýmovou spolupráci vzhledem k nutnosti koordinovat rozhraní a jeho změny. [50]

3.2 Systémy využívané v právních kancelářích

V této kapitole se budeme zabývat existujícími informačními systémy a digitálními službami, které jsou k dispozici pro advokátní kanceláře či samostatné advokáty v České republice. Řadí se sem systémy, které jsou využívány advokátními kanceláři pro zefektivnění jejich chodu a organizace a zlepšení úrovně jimi poskytovaných služeb klientům. Mezi tyto systémy patří první specializované nástroje a informační systémy přímo pro advokátní kanceláře. Ty nabízejí funkcionalitu jimi nejčastěji vyžadovanou a jsou úzce spjaty s touto doménou. Nejčastěji sem patří funkcionalita pro správu klientů advokátní kanceláře a organizaci spisů patřící k řešeným případům. Další funkce mohou být spojeny s vyhledáváním právních informací, s hlídáním lhůt či funkce pro zjednodušení účtování provedené práce. Důležitým aspektem je také integrace s veřejnými rejstříky popsány v první části analýzy a další napojení na externí systémy.

Za druhé jsou to více univerzální softwarové nástroje, které mohou být využívány pro organizaci v libovolné malé až střední firmě. Ty typicky splňují obecné požadavky takovýchto společností a mohou nabízet spoustu funkcí, kterých v advokátní kanceláři nebude využito, nebo naopak mohou postrádat funkce specificky vyžadované v právní doméně. Běžně nabízejí funkce pro obecnou správu klientů či funkce pro přidělování úkolů zaměstnancům a organizaci práce v týmu.

V neposlední řadě jsou využívány také menší nástroje a služby, sloužící jednomu specifickému účelu, často zvolené pro doplnění funkcionality chybějící u primárně používaného systému. Do této kategorie mohou patřit například nástroje na zaznamenávání a vykazování odvedené práce zaměstnanců, komunikační platformy a další vedlejší nástroje.

Systémy by bylo možné dále rozdělit na základě jejich přístupu k implementaci a architektuře. Hlavním rozdílem potom je, zda se jedná o desktopové aplikace, které jsou instalovány na počítačích lokálně, či jsou dostupné jako webové aplikace skrz prohlížeč a není tak nutné je instalovat. U desktopových aplikací může být ještě rozdíl v provozu systému a ukládání dat, tedy zda systém běží na lokálním serveru a data jsou ukládána primárně lokálně v rámci advokátní kanceláře (tzv. on premise verze), či je využíváno cloudových služeb, jako je tomu v případě webových aplikací.

Analyzované systémy celkově nabízí propracovanou funkcionalitu a kromě vybraných a popsanych funkcí všechny nabízí standardní funkce informačních systémů jako je například přihlašování a autorizace uživatelů, nastavení pravomocí uživatelů a další běžné funkce.

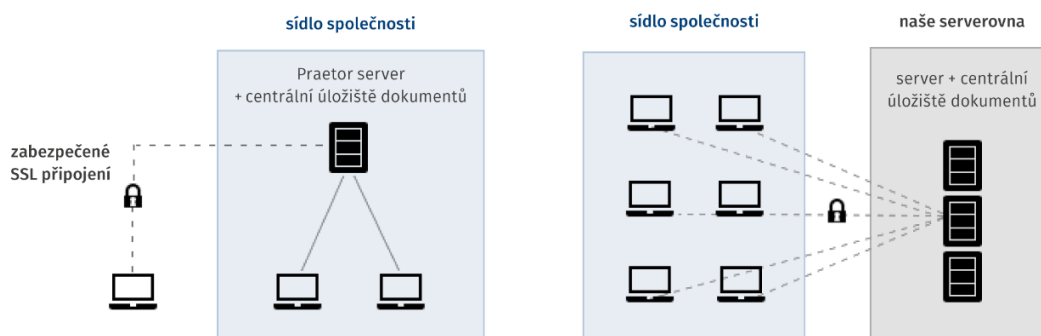
3.2.1 Praetor

Jedním z používaných systémů v advokátních kancelářích je systém Praetor¹. Podle jeho tvůrců je systém vyvíjen s advokáty, díky čemu není zbytečně složitý a nabízí přesně to co advokátní kanceláře potřebují. Popisují ho jako *“přehledný, intuitivní, snadno ovladatelný inovativní systém pro úsporné a efektivní řízení advokátní kanceláře a zvládnutí právní*

¹<https://www.wolterskluwer.com/cs-cz/solutions/praetor>

agendy.” Systém je dodáván společností Wolters Kluwer ČR, a.s. což je globální společnost působící jako poskytovatel odborných informací, softwarových řešení a služeb mimo jiné v právním, zdravotním, a finančním sektoru. [23]

Systém je koncipovaný primárně jako desktopová aplikace, dostupná pro platformy windows a mac OS. Je nabízena ale i mobilní verze. Podle webu poskytovatele je systém postavený na základě architektury klient-server, přičemž se uživatelé mohou rozhodnout, zda systém chtějí hostovat lokálně na vlastním serveru či využijí cloudového řešení. Možnost s využitím vlastního serveru je prezentovaná jako defaultní varianta.[5] Systém působí profesionálně a nabízí rozsáhlou a komplexní funkcionalitu. Mezi nabízené funkce patří mimo jiné následující:



Obrázek 3.2: Pohled na schémata možných nasazení systému Praetor. Převzato z webu poskytovatele: [<https://www.wolterskluwer.com/cs-cz/solutions/praetor/architektura>]

Spisová evidence

Systém nabízí správu spisů, což je jedna ze zásadních funkcionalit, která je vyžadována od každého systému pro právní kanceláře. Spisy je možné filtrovat a řadit na základě klientů, protistran, kategorií a dalších kritérií. Ve spisech je možno též vyhledávat. Ke spisu je vždy vázáno vše důležité jako související dokumenty, činnosti, lhůty a pošta.

Správa dokumentů

Dokumenty mohou být nahrávány ke spisu a jsou automaticky číslovány, stejně tak jejich následující možné verze. Systém obsahuje též šablony pro automatické vytváření některých typů dokumentů.

Napojení na rejstříky

Systém automaticky získává informace o klientech a protistranách z obchodního rejstříku, zároveň kontroluje zda subjekty nejsou v insolvenční skrzi insolvenční rejstřík a jsou sledovány i případné změny s možností nastavení upozornění.

Systém také získává informace z aplikace infoSoud o průběhu řízení a o zúčastněných a rozhodujících orgánech u řízení na základě spisové značky. Spojení se systémem UIDR umožňuje ověření existence poštovních adres.

Databáze subjektů a kontaktů

K subjektům je možné ukládat více kontaktů včetně jejich vztahu k případu a jejich preferovaného přiřazení ke spisům.

Správa pošty

Systém nabízí přehled veškeré korespondence prostřednictvím emailových zpráv, datových zpráv i listovních zásilek, které byly elektronicky zaevidovány. Je možné přímo automaticky stahovat a zasílat datové zprávy i emaily. Dokumenty mohou být zároveň připraveny k odeslání jako listovní zásilka.

Ekonomické funkce

Systém nabízí funkcionalitu pro zjednodušení vykazování práce a fakturace. Je možné automaticky generovat a zasílat faktury a zobrazit si celkový přehled vyúčtování.

Další funkce

Systém dále umožňuje například elektronické podepisování, napojení na účetní programy, a také rozšíření o další extra moduly a funkce. Mezi ty vybrané patří advokátní úschovna, autorizovaná konverze dokumentů, moduly pro pohledávky či agendu nemovitostí a mnohé další.

Mezi některé technologie využité pro implementaci patří Microsoft .NET Framework 4, Microsoft Sql Server 2012 a komponenty DevExpress. [23]

3.2.2 Ispis

Dalším dostupným systémem je služba s názvem *iSpis*² nebo InteligentníSpis který je webovým softwarem zaměřeným na právní problematiku vyvíjen firmou Sokordia, s.r.o.. Podle webových stránek je určen pro finanční společnosti, advokáty apod. a mezi jeho zmíněné přednosti patří podle poskytovatelů **minimalizace finančních rizik, zefektivnění vymáhání pohledávek a ulehčení práce právníkům**. Nabízí odděleně dvě služby, z nichž jedna je nazvána *Advokátní spis* a bude dále popsána. Druhou službou je software pro správu a vymáhání pohledávek. [16] Jelikož se jedná o webovou aplikaci, která podporuje responzivní design, je aplikace multiplatformě dostupná na libovolných zařízeních s podporou webového prohlížeče, včetně mobilních zařízení. Systém nabízí následující vybranou funkcionalitu:

Evidence případů a subjektů

Funkcionalita pro správu spisů advokátní kanceláře a souvisejících subjektů. Umožňuje evidovat stav případu, dokumenty a přidávat úkoly či poznámky. Umožňuje automatické předvyplnění některých informací díky napojení na rejstříky.

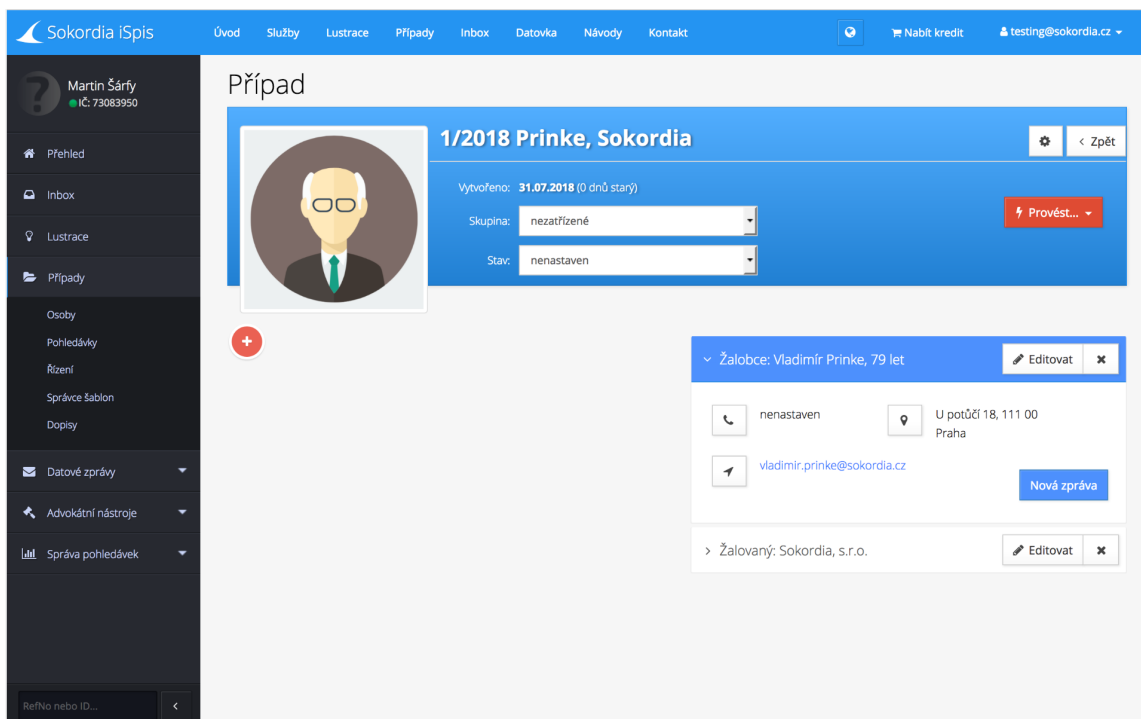
Napojení na rejstříky

Systém umí získávat informace o subjektech z obchodního rejstříku a předvyplnit tak zadávané údaje. Je také možné provést lustraci osoby/firmy a ověřit zda není v exekuci skrz centrální evidenci exekucí České republiky. Pomocí napojení na insolvenční rejstřík je možné monitorovat i případné změny v solvenci subjektů. Systém sleduje změny jednání skrze *infoSoud* a *infoJednání* na základě spisové značky.

Správa pošty

Jedná se jak o evidenci datových zpráv tak možnost nahrání a evidence pošty fyzické s informací o odeslání či přijetí. Je možné automaticky odesílat i fyzickou poštu prostřednictvím využití služby *dopisOnline* od české pošty.

²<https://ispis.cz/>



Obrázek 3.3: Snímek obrazovky s pohledem na stránku případu v systému Ispis. Převzato z webu poskytovatele: [<https://ispis.cz/>]

Elektronické platební rozkazy

Systém umožňuje vygenerovat a odeslat návrh na vydání elektronického platebního rozkazu. Po odeslání platebního rozkazu systém zaeviduje číslo účtu a částku pro zaplacení, přidělenou spisovou značku podání a následně podání kontroluje na *infoSoudu*.

Úkoly a úkony

V rámci spisu je možné vytvářet úkoly a přidělovat je jiným zaměstnancům. Veškeré úkoly jednoho zaměstnance se poté zobrazují v jeho inboxu spolu s datovými zprávami, emaily a dalšími aktualitami. Podobně je možné přidávat úkony s jejich cenou na základě kterých je později možné vystavit fakturu.

Hlídní a převod nemovitosti

U případů je možné evidovat nemovitosti, kterých se případ týká. U nemovitosti je pak zobrazen její výpis z katastru nemovitostí a je možné nastavit hlídání změn a upozornění na ně. K dispozici je také funkcionality pro zjednodušení převodu nemovitosti. Na základě zadaných údajů o nemovitosti a nabyvateli je skrz průvodce možné vytvořit návrh na vklad práva do katastru nemovitostí a související smlouvy.

Další funkce

V systému je zabudovaný jednoduchý textový editor, pomocí kterého je možné vytvářet různé propisovatelné šablony dokumentů, s nimiž mohou být dokumenty později generovány.

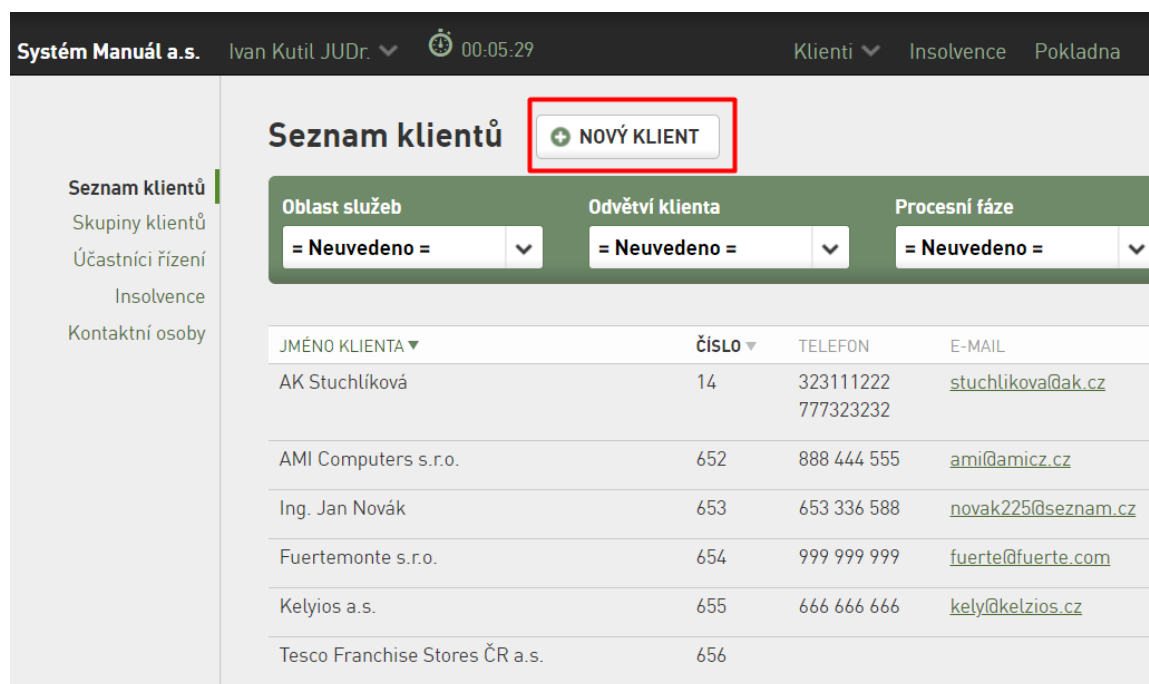
Systém *Ispis* je jednodušší než systém *Preator* a nabízí méně možností u jednotlivých funkcí. Může tím tak být jednodušší na používání pro uživatele. Hlavní funkcionality ale nabízí obdobnou. Uživatelské rozhraní je moderní a působí přívětivě. V některých případech však může být trochu matoucí. Systém je totiž zabudovaný “přímo” ve webové stránce, která systém prezentuje a popisuje. Mezi nabízenými funkcemi systému v hlavní navigační

lišťe je tak pod některými položkami například místo očekávané funkcionality jen její popis a odkázání na další stránky kde je možné ji najít apod. Namátkově také některé prvky a funkce při testovacím používání aplikace nefungovaly/zobrazovaly nečitelné systémové chybové hlášky. Cena systému *Ispis* činí 5 500,- Kč za rok. Avšak v prvním roce používání je tato služba poskytována zcela zdarma. [16]

K implementaci systému bylo využito technologií *node.js*, *React*, frameworku *next.js*, *jquery* a UI frameworku *bootstrap*.

3.2.3 Isak

Systém *Isak*³ patří mezi další vybrané systémy k analýze. Je prezentován jako “*software pro přehledné a bezpečné vedení advokátní kanceláře*”. [17] Nabízí řešení pro malé a velké advokátní kanceláře. V tomto případě se jedná opět o webovou aplikaci, je tak multiplatformně dostupná. Funkcionality nabízí v zásadě srovnatelnou s předchozími dvěma řešeními a je napojen na všechny zmiňované rejstříky. Jsou zde však jisté odlišnosti. Rozdíly jsou také ve způsobu zpřístupnění funkcionalit prostřednictvím prvků uživatelského rozhraní. Dále jsou tak popsány jen vybrané hlavní funkce, které mají být implementovány i v rámci vytvářeného systému a funkce které jsou něčím zajímavé nebo výrazněji odlišné od předchozích systémů.



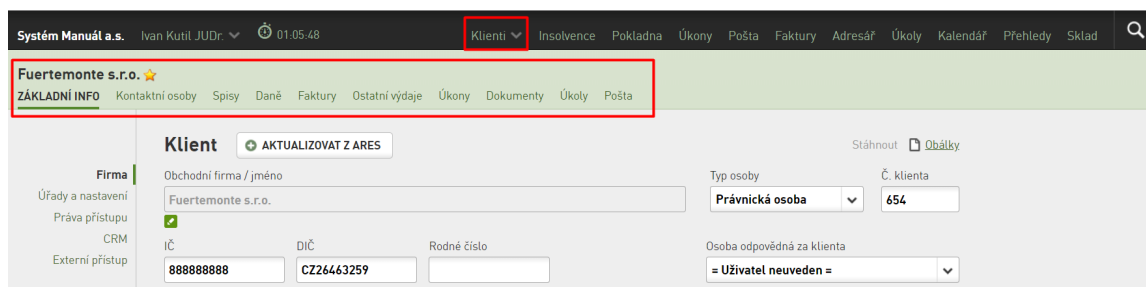
Obrázek 3.4: Snímek obrazovky s pohledem na stránku klientů v systému Isak. Převzato z webu poskytovatele: [<https://dokumentace.isak.cz/doku.php/>]

Evidence klientů s spisů

Systém opět nabízí propracovanou funkcionality pro správu klientů a jejich spisů. Umožňuje globální pohled na všechny klienty. Seznam klientů i spisů je zobrazen formou tabulky, ve které je možné řadit a filtrovat podle jednotlivých sloupců a dalších kritérií.

³<https://www.isak.cz/>

Po detailním zobrazení vybraného klienta je možné zobrazit si jeho spisy, kontaktní osoby, dokumenty a další, skrze horní navigační lištu. Spisy jsou dále rozdělené do dvou záložek jako aktivní a archivované.



Obrázek 3.5: Snímek obrazovky s pohledem na stránku klienta a navigaci v systému Isak. Převzato z webu poskytovatele: [<https://dokumentace.isak.cz/doku.php/>]

Správa pošty

Agenda pošty opět umožňuje globální evidenci veškeré doručené a odeslané pošty a to ve třech způsobech doručení jako u předchozích systémů:

- poštu doručovanou běžnou poštovní přepravou,
- datové zprávy doručované a odesílané prostřednictvím datových schránek,
- zprávy elektronické pošty (e-mailly).

Pokud má záznam pošty přiřazeného klienta a spis, promítne se automaticky také v přehledu pošty příslušného klienta a spisu.

Úkony a fakturace

Systém podporuje vytváření úkonů na základě kterých je možné vytvářet faktury pro klienty. Úkony je možné v systému vytvářet flexibilně na několika místech, jmenovitě z přehledu úkonů, přímo u klientů a spisů, nebo z úkolu či stopek.

Pro fakturaci systém nabízí komplexní funkcionalitu s mnohými nastaveními. Na základě vybraných podkladů mezi které patří úkony, ale i paušály a další položky. Fakturovat lze pro klienta, skupiny klientů nebo na základě jednoho spisu. Dále je možné upřesnit více detailů jako daně, poplatky, možné slevy apod.

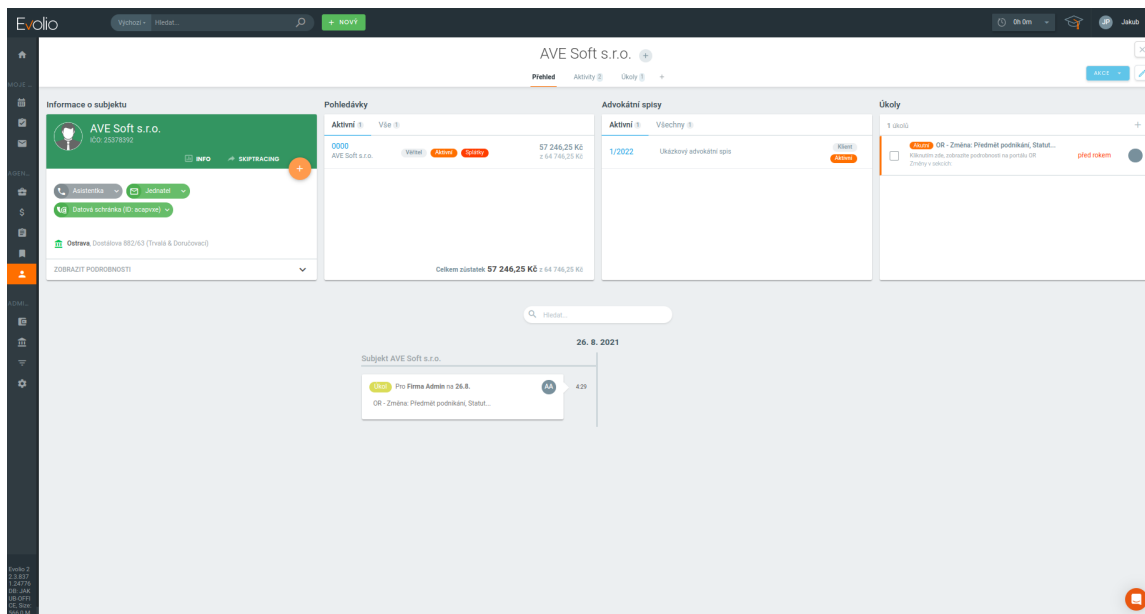
Systém nabízí celkově propracovanou funkcionalitu a k dispozici je výčet dalších funkcí z nichž některé jsou dostupné za další příplatky. Systém je dostupný ve 3 standardních balíčcích pro malé (od 150 Kč/měsíc), střední (od 1500 Kč/měsíc) a velké advokátní kanceláře, kde možnosti a cena poslední možnosti jsou individuální. Cena se dále odvíjí od počtu uživatelů systému.

3.2.4 Evolio

Posledním vybraným systémem k analýze je systém *Evolio*⁴, prezentovaný jako “*chytrý software pro každou advokátní kancelář pro automatizované vedení elektronického spisu, inteligentní vykazování práce či hromadné podávání EPR na jeden klik*” [12]. Systém nabízí obdobnou funkcionalitu, jako předchozí systémy opět s rozdíly v uživatelském rozhraní,

⁴<https://evolio.cz/>

kteře působí asi nejmoderněji z vybraných systémů a v propracovanosti některých funkcí. Klade větší důraz na celkový přehled a historii spisů a také nabízí extra funkce pro konfigurovatelnou automatizaci některých úkonů. Systém je opět koncipován jako webová aplikace. Hlavní funkcionalita systému je přístupná prostřednictvím postranního navigačního panelu a související funkce/kategorie je následně možné zobrazit jako podstránky pomocí horní navigační lišty.



Obrázek 3.6: Snímek obrazovky s pohledem na stránku klienta v systému Evolio. Převzato z webu poskytovatele: <https://evolio.cz/>

Přehled subjektů

Pod záložkou subjektů je přístupná tabulka naposledy zobrazených subjektů. Jako subjekty v systému mohou vystupovat jak klienti advokátní kanceláře, tak potenciální protistrany, či jakékoliv další zúčastněné strany. Subjektem může být jakákoli fyzická osoba, fyzická podnikající osoba či právnická osoba. Zajímavou vlastností je možnost “zobrazit všechny subjekty” se kterou je načten seznam právnických a podnikajících osob. Data jsou s největší pravděpodobností agregovaná z z obchodního a živnostenského rejstříku. U těchto subjektů je možné zobrazit detail a přímo je používat jako subjekt v rámci systému se všemi náležitými možnostmi jako u ručně vytvořeného subjektu.

Detailní zobrazení subjektu nabízí přehled všech náležitostí na jedné stránce, odkud je možné zobrazit konkrétní agendy (aktivity, spisy, dokumenty) pomocí horní navigace. V přehledu je k dispozici časová osa zobrazující historii aktivit spojených s klientem.

Přehled spisů

Přehled všech spisů v systému je možné si zobrazit klasicky v podobě tabulky, nebo rozdělené do panelů na základě kategorie do které je spis přiřazen, štítku, či stavu ve kterém se spis nachází (aktivní či archivované). Možnost tohoto zobrazení se liší od předchozích systémů. Spisy je také možno filtrovat podle rozličných kritérií.

Detailní zobrazení spisu je velmi podobné detailu subjektu. Na hlavní stránce je k vidění přehled se všemi náležitostmi spisu, jako je klient, další subjekty, úkoly spojené se

spisem a časová osa aktivit. Odtud je možné zobrazit záložky pro finance, aktivity, úkoly a dokumenty.

Pošta, výkaz práce a faktury

Správa pošty nabízí opět stejné tři zdroje zpráv jako předchozí systémy. Nové datové zprávy a emaily je možné stáhnout na vyžádání.

Pro výkaz práce je k dispozici týdenní přehled pro zapisování aktivit. Ty je možné přidat opět z více míst, jako například z dokončeného úkolu. Výkaz slouží jako základ pro vytváření faktur pod záložkou vyúčtování, kde najdeme celkový finanční přehled. Při tvorbě faktury je možné vybrat položky ze seznamu nevyfakturovaných činností a zahrnout je do vytvářené faktury. Výslednou fakturu lze poté vygenerovat jako pdf dokument. K dispozici je též více detailních nastavení ohledně vyúčtování a fakturace.

Další funkce

Samozřejmostí jsou další funkce jako napojení na rejstříky, kalendář, plánování událostí, hlídání lhůt, či funkce pro automatizace a další.

3.3 Vybrané implementační technologie

V této kapitole se zaměříme na obecné seznámení s hlavními z vybraných technologií použitých při implementaci samotného systému. Každá technologie je stručně představena včetně hlavních výhod jejího použití. Bližší odůvodnění použití některých technologií je přiblíženo v části popisu implementace.

3.3.1 Node.js

Node.js⁵ je open-source, multiplatformní prostředí sloužící pro vývoj serverové části aplikací. Je založen na JavaScriptovém runtime prostředí V8 od společnosti Google a umožňuje psát serverový kód pomocí jazyka JavaScript, který byl původně určený pro běh v prohlížeči na straně klienta. Mezi hlavní výhody Node.js patří asynchronní, událostmi řízený model, který umožňuje vysokou propustnost a dobré škálování aplikací a také možnost sdílení a znovupoužití částí kódu mezi klientem a serverem, což může přispět k rychlejšímu vývoji a jednodušší údržbě. [1]

3.3.2 Vue.js

Vue.js⁶ je progresivní frontend framework pro tvorbu uživatelských rozhraní ve webovém prostředí. Umožňuje tvorbu znovupoužitelných komponent, přináší deklarativní přístup k manipulaci s DOM a implementuje tzv. reaktivní systém pro sledování změn stavu aplikace. Aktualizace DOMu je založena na principu virtuálního DOMu, který efektivně porovnává jeho nový a starý stav a ve skutečném DOMu aplikuje pouze nezbytné změny. Je zajištěno také seskupení jednotlivých operací. Tento přístup tak zvyšuje výkon aplikace oproti provádění operací přímo nad reálným DOMem. [57]

Reaktivní systém umožňuje automatickou aktualizaci komponent při změnách stavu dat. Tento systém sleduje závislosti mezi daty a komponentami a zajistí, že změny v datech se okamžitě promítnou do zobrazení, bez nutnosti manuální aktualizace. Zajišťuje tak jednodušší a intuitivnější způsob práce se stavem aplikace. [26]

⁵<https://nodejs.org/en>

⁶<https://vuejs.org/>

3.3.3 Nuxt.js

Nuxt.js⁷ je vysokoúrovňový framework operující nad Vue.js, který zjednodušuje vývoj univerzálních aplikací, pro které je důležitá i serverová část a je navržen pro použití s Node.js. Definuje adresářovou strukturu projektu a poskytuje automatické konfigurace, které zahrnují například automatické generování url cest, zjednodušení vytváření API koncových bodů, funkce pro jednoduchou navigaci mezi stránkami. Může také přispět ke zlepšení výkonu aplikace pomocí možnosti renderování frontend komponent již na straně serveru. [20]

3.3.4 PostgreSQL

PostgreSQL⁸ je open-source objektově-relační databázový systém. To znamená, že kromě tradiční tabulkové struktury využívané v čistě relačních databázových systémech jako je např. MySQL, nabízí rozšíření o objektově orientované koncepty, jako jsou dědičnost, funkce a složitější datové typy, jako jsou např. JSON a JSONB pro ukládání a manipulaci s JSON daty. Paří do rodiny SQL databázových systémů a podporuje tak standardní SQL dotazy, ale zároveň přináší rozšiřující vlastnosti. [2]

3.3.5 Prisma

Prisma⁹ je ORM (Object-Relational Mapping) nástroj, který se zaměřuje na zjednodušení práce s databází v prostředí Node.js. Nabízí generované API, které se přirozeně mapuje na strukturu databáze a umožňuje pracovat se získanými daty přímo jako s objekty v JavaScriptu se specifikovanými datovými typy. Umožňuje jednoduše vytvářet, číst, aktualizovat a mazat záznamy v databázi, bez nutnosti využití SQL dotazů pro běžné operace, podporuje ale i tvorbu vlastních SQL dotazů. [24]

⁷<https://nuxt.com/>

⁸<https://www.postgresql.org/>

⁹<https://www.prisma.io/>

Kapitola 4

Specifikace a návrh systému

Tato kapitola se zabývá procesem tvorby návrhu nového informačního systému pro advokátní kanceláře s agregací dat z veřejných databází. Kapitola navazuje na předchozí kapitolu analýzy, ve které byly získány teoretické znalosti jako základ pro tuto kapitolu a je z nich dále vycházeno. Cílem návrhu bude mimo jiné identifikovat a specifikovat klíčové funkce, které bude systém nabízet, a určit jak tyto funkce zpřístupnit uživateli, aby jeho práce mohla být efektivnější. Během návrhu je využíváno UML modelování pro vytvoření jednoznačného popisu dále využitelného při implementaci systému. [43] V první části jsou identifikovány potřeby cílových uživatelů na základě poznatků získaných analýzou procesů advokátní kanceláře v kapitole 2.3. Podle nich a podle všeobecných požadavků daných zadáním je sestavena detailní specifikace veškerých požadavků na vytvářený systém. To zahrnuje identifikaci aktérů vystupujících v systému a možných případů užití, které aktéři v systému vykonávají. Dále je navržena část zabývající se získáváním dat z veřejných databází s přihlédnutím k formátu dat a omezením těchto zdrojů. S tím je spojen návrh databáze pro ukládání a organizaci těchto a dalších ukládaných dat. Následně je proveden iterativní návrh uživatelského rozhraní pro vhodné zobrazení dat figurujících v systému a umožnění intuitivní práce se systémem. Na závěr je navržena celková architektura systému vycházející z architektonického vzoru MVP.

4.1 Specifikace požadavků na systém

Cílem práce je vytvoření informačního systému pro advokátní kanceláře, který bude napojený na veřejné databáze, automaticky z nich získávat relevantní informace a ukládat je. Na základě analýzy procesů advokátní kanceláře, a průzkumu existujících řešení je možné stanovit, jaké funkce by systém měl nabízet, jaké činnosti budou uživatelé v systému vykonávat nejčastěji a jaká data bude potřeba ukládat. Toto je popsáno formou neformální specifikace, která vychází z analýzy zadání práce a vybraných podkapitol v kapitolách 1 a 2. Na základě neformální specifikace jsou dále identifikováni důležití aktéři vystupující v systému a je navržen diagram případů užití.

4.1.1 Neformální specifikace

Systém bude primárně využíván **menšími advokátními kancelářemi** nebo **samostatnými advokáty**. Využitelný by mohl být též společnostmi nabízející účetní služby, či vy-
máhání pohledávek, jejichž agenda se v nějakých ohledech překrývá s agendou advokátních společností. V dalším popisu bude však kladen důraz na advokátní kanceláře. Uživateli sys-

tému budou tedy primárně **zaměstnanci advokátní kanceláře** na všech úrovních - od vedení advokátní kanceláře, přes jednotlivé advokáty po praktikanty a asistenty. Pro advokátní kancelář tak bude důležité **spravovat přístup zaměstnanců** do systému s možností stanovení rolí a zneprístupnění některých částí systému pro uživatele bez udělených oprávnění, stejně tak zpřístupnění různých funkcí systémů by mělo být konfigurovatelné pomocí **přidělování rolí** jednotlivým zaměstnancům.

Advokátní kancelář a advokáti poskytují své služby klientům. Stávající i noví potenciální klienti je pravidelně kontaktují se svými požadavky skrze různé kanály jak bylo popsáno v kapitole 2.3.1. V zájmu kanceláře je získávat co nejvíce nových klientů, udržet si stávající klienty a zároveň stále nabízet vysokou kvalitu poskytovaných právních služeb. Proto je pro kanceláře důležité mít dobrý **přehled o všech svých klientech** a kontaktu s nimi, stejně tak jako o nových potenciálních klientech pro zvýšení šance jejich akvizice (jejich získání, uzavření smlouvy o poskytování právních služeb). S tím spojené jsou informace o tom, v jakém stavu se jednání s klientem nachází, případně jaké akce již byly provedeny, případně co bylo s klientem již probráno a dohodnuto. Systém by tak měl podporovat **efektivní správu klientů** a možnost zaznamenávání vlastních informací s nimi souvisejících. Klientem mohou být fyzické osoby, fyzické podnikající osoby a právnické osoby. O těchto osobách je možné **získat užitečné informace** z veřejných rejstříků, jak bylo přibliženo v kapitole 2.1. Právníci s těmito informacemi často pracují hned od prvního kontaktu s potenciálním klientem. Systém by tak měl umožňovat **automatické a jednoduché získání** těchto dostupných informací z **externích zdrojů**. Jelikož se tyto informace mohou v průběhu času měnit, systém by měl **pravidelně kontrolovat aktuálnost** těchto informací. Aby případná změna nebyla přehlédnutá, měl by na ni **systém upozornit** odpovídající zodpovědnou osobu. Ke klientovi se též může vztahovat více kontaktních osob, které je také užitečné v systému evidovat.

S každým získaným klientem se váže potřebná **dokumentace**. Od smlouvy o spolupráci, přes plnou moc po veškeré další dokumenty související s klientem a s ním řešenými případy. Systém tak musí podporovat **ukládání dokumentů** souvisejících s klienty. **Dokumenty je nutné organizovat** a mělo by tak být možné dokumenty **seskupovat do vlastních složek** a podsložek. Dokumenty se také často aktualizují a mohou se vyskytovat ve více verzích. Mělo by tak být možné vytvářet **více verzí stejného dokumentu**, se zachováním verzí předchozích.

Advokátní kanceláře mohou s každým ze svých klientů řešit několik různých případů s odlišným průběhem, zodpovědnostmi a dokumenty. Pro každého klienta by tak mělo být možné **vytvářet jednotlivé spisy** pro různé případy. Do spisu případu je opět potřeba nahrávat **dokumenty související s případem** a jejich nové verze. Tyto dokumenty musí být taktéž dále **organizovatelné do složek** podle vlastní potřeby. Každý řešený případ může být součástí (soudního) řízení. V takovém případě je k případu vázána **spisová značka** přidělená některým soudem, kterou je možné ke spisu přidat. Systém bude poté **automaticky získávat a ukládat důležité informace o řízení** a jeho změnách a o těchto skutečnostech **informovat odpovídající osoby**. S případem typicky souvisí **protistrana**, proti které je řízení vedeno, či jiné **zainteresované subjekty** související s případem, které je třeba **ve spise evidovat**, včetně případných souvisejících **kontaktních osob** ke každému subjektu.

V rámci klienta či konkrétního případu zaměstnanci **vykonávají určité činnosti a právní úkony**, na základě kterých je poté klientovi účtován poplatek za poskytnuté služby. Úkon typicky trvá nějaký čas práce, spadá do některé kategorie poskytovaných služeb a může být později vyúčtován. Zaměstnanci by tak měli mít možnost **zaznamenávat své**

činnosti provedené pro klienta, či v rámci spisu klienta, ale i na úrovni celé advokátní kanceláře. Tyto záznamy budou zároveň sloužit jako **přehled jimi vykonané práce** a jako podklad pro **vytvoření faktury** pro klienta. Pověřený pracovník může následně vybrat několik úkonů k vyúčtování pro klienta a na základě nich systém automaticky **vygeneruje fakturu**. Tyto úkony jsou dále označeny jako vyúčtované.

Jelikož advokátní kanceláře stráví hodně času komunikací s klienty, úřady, soudy a dalšími orgány je pro ně též důležité mít veškerou korespondenci dobře organizovanou. Systém by tak měl umožnit **správu elektronické a poštovní komunikace**.

V rámci technických požadavků nejsou kladené speciální nároky na systém. V potaz jsou tak brány všeobecné požadavky. Žádoucí je aby systém byl multiplatformní a dostupný na různých zařízeních a musí umožňovat spolupráci více lidí zároveň, kteří se k systému mohou připojovat z různých míst. Očekává se také připojení k internetu, jelikož má systém pracovat s externími zdroji dat, které je potřeba udržovat aktuální.

4.1.2 Aktéři v systému a jejich role

Na základě neformální specifikace a bližší analýzy je možné identifikovat jednotlivé aktéry, kteří v rámci systému budou vystupovat a jaké budou jejich role. Také jakým činnostem budou systému využívat a jaké by tak měly být funkce, které to budou podporovat. Je možné nastínit očekávané pravomoci, které by však měly být přizpůsobitelné každou advokátní kanceláří.

Vedoucí advokátní kanceláře

Může se jednat o jednu nebo více osob zodpovědných za vedení advokátní kanceláře a mající tak na starost většinu klíčových rozhodnutí. Tato osoba pravděpodobně rozhodne o zavedení systému do advokátní kanceláře a bude se účastnit klíčových nastavení systému.

Měla by tak mít možnost zpřístupnit systém vybraným zaměstnancům a udělovat jim pravomoci v systému, včetně přidání dalších vedoucích pracovníků. V neposlední řadě mohou mít na starost i vyúčtování služeb pro klienty. Mají tak přístup k účtování a tvorbě faktur. Defaultně by měla mít přístup ke všem ostatním částem a funkcím systému.

Advokát

Jedná se o hlavní zaměstnance advokátní kanceláře, kteří budou systém používat pro svou každodenní agendu. Všeobecně je možné říci, že se účastní většiny každodenních procesů v advokátní kanceláři souvisejících s poskytováním právních služeb. Mají na starost získávání a evidenci klientů, vytváření a správu jejich spisů a vedení dokumentace. Dále komunikují s klienty, úřady a dalšími entitami a korespondenci evidují. Musí tak mít přístup k funkcionalitě k tomu potřebné.

Asistent

Pod tímto aktérem si je možné představit v podstatě všechny další zaměstnance advokátní kanceláře, kteří mají mít přístup do systému. Spadají sem skuteční asistenti, kteří například nahrávají skenované dokumenty a poštovní korespondenci ke spisům. Pod aktéra je též možné zahrnout koncipienty a právní praktikanty, kteří pracují s existujícími klienty a spisy, mohou zaznamenávat související a provedené činnosti či k nim nahrávat dokumenty, avšak neměli by pravděpodobně mít pravomoc měnit či mazat klienty a spisy samotné, což je primárně činnost nadřízených advokátů. Podobně nebudou mít na starost ani účtování provedené práce.

Nepřihlášený uživatel

Jedná se o nového (potenciálního) uživatele systému, který k němu ještě nemá přístup. Může se přihlásit, svými přihlašovacími údaji pokud mu byl přístup již dříve přidělen jako zaměstnanci advokátní kanceláře, nebo může zažádat o registraci pro vytvoření a inicializaci nové instance systému pro své použití např. ve své advokátní kanceláři.

Čas/Notifikátor

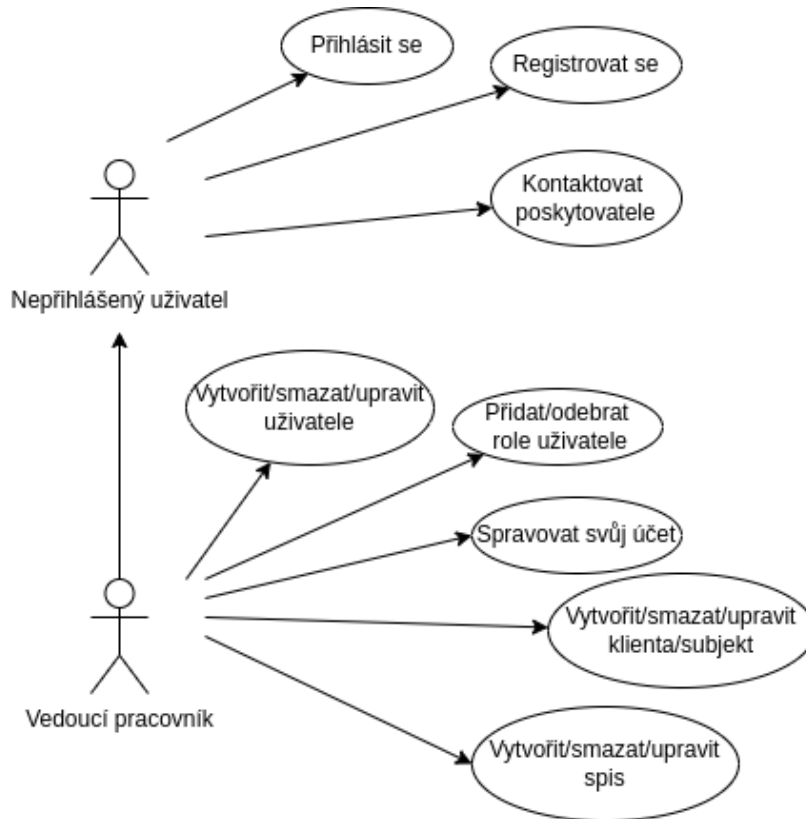
Jedná se o systémového aktéra, který bude vytvářet upozornění pro uživatele na základě hlídaných lhůt, či nastalých změn v externích systémech odkud budou agregována aktuální data.

4.1.3 Případy užití

V této části je navržen diagram případů užití. Jedná se o UML diagram sloužící k modelování chování, který graficky znázorňuje veškeré funkce přístupné jednotlivým aktérům systému. Blíže také znázorňuje vztah mezi některými funkcemi pomocí vazeb extend a include. Diagram přehledně znázorňuje dostupnou funkcionalitu a hierarchii pravomocí v systému, způsobem, který je snadno pochopitelný i osobami bez technického vzdělání. Je tak často využíván při konzultaci specifikací se zákazníky. Umožňuje navíc rozdělení vývoje systému do více fází na základě vybrané funkcionality, která bude v každé fázi implementována. Vzhledem ke komplexnosti vytvářeného systému bude vývoj rozdělen do dvou iterací. To umožní jednodušší stanovení cílů během vývoje a také získání průběžné zpětné vazby od uživatelů. [43]

Diagram první iterace

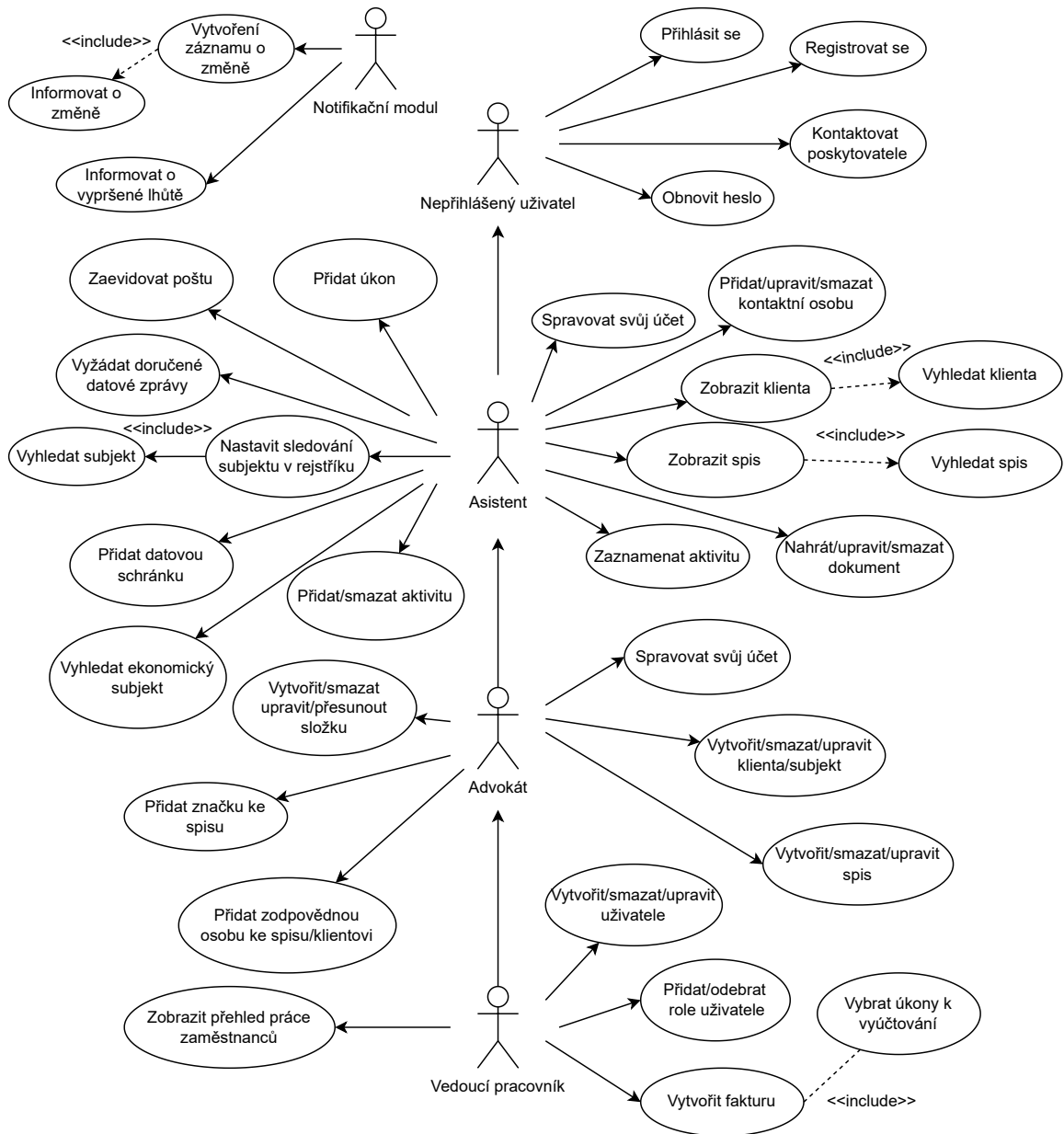
Iterace bude zaměřená na implementaci základní autentizace uživatelů a zabezpečení přístupu do systému. S tím spojená registrace nových uživatelů a přihlašování do systému pro existující uživatele. Defaultním uživatelem potom bude vedoucí pracovník. Ten bude moci do systému přidávat další uživatele a spravovat jejich role a dále vytvářet a spravovat nové klienty a spisy. Touto funkcionalitou je možné demonstrovat základní funkcionalitu systému. Bude sloužit jako základ pro přidání zbývajících aktérů a implementaci dalších rozšiřujících funkcí v druhé iteraci.



Obrázek 4.1: Diagram případů užití pro první iteraci.

Diagram druhé iterace

Během druhé iterace bude již implementována kompletní funkcionalita systému. Toto bude zahrnovat veškeré funkce vyplývající z neformální specifikace požadavků.



Obrázek 4.2: Diagram případů užití pro druhou iteraci s většinou případů užití.

4.2 Napojení na veřejné databáze

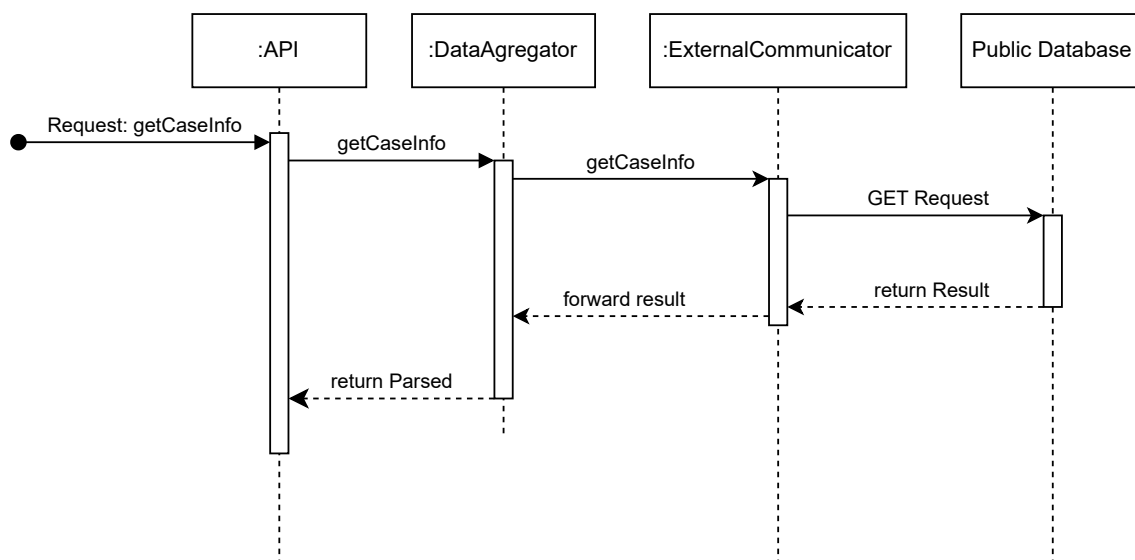
V této části je navržen modul pro získávání a zpracování dat z externích zdrojů popsaných v úvodní kapitole, s přihlédnutím ke struktuře dat a limitacím těchto zdrojů. Požadavkem na modul je, aby získával a zpracovával data z různých zdrojů nezávisle a data zpracoval a poskytoval v jednotném formátu. Změna formátu dat z některého ze zdrojů by neměla vést k selhání celého modulu a nutnosti jeho celého předělávání. Pro účely systému bylo

vybráno několik dále zmíněných externích zdrojů informací jako nejvíce relevantních a byly uvažovány při tvorbě následujícího návrhu získávání a zpracování dat z nich.

4.2.1 Základní informace o subjektech

Jak bylo podrobně popsáno v kapitole 2.1, základní informace o subjektech lze získat z několika veřejných rejstříků. Obchodní rejstřík slouží jako zdroj informací o právnických osobách, včetně jejich názvů, sídel, identifikačních čísel a dalších důležitých údajů. Na druhé straně živnostenský rejstřík poskytuje podobné informace o fyzických podnikajících osobách, které provozují živnostenskou činnost.

Přístup k těmto informacím je klíčový pro právníky při zahájení spolupráce s klienty, při provádění analýzy či při tvorbě smluv. Systém by měl umožnit snadný přístup k těmto datům a získávat je tak automaticky. Následující sekvenční diagram znázorňuje možný průběh získání dat včetně nastínění modulů, které pro to budou vyžadovány.



Obrázek 4.3: Sekvenční diagram znázorňující proces získání dat z externích systémů.

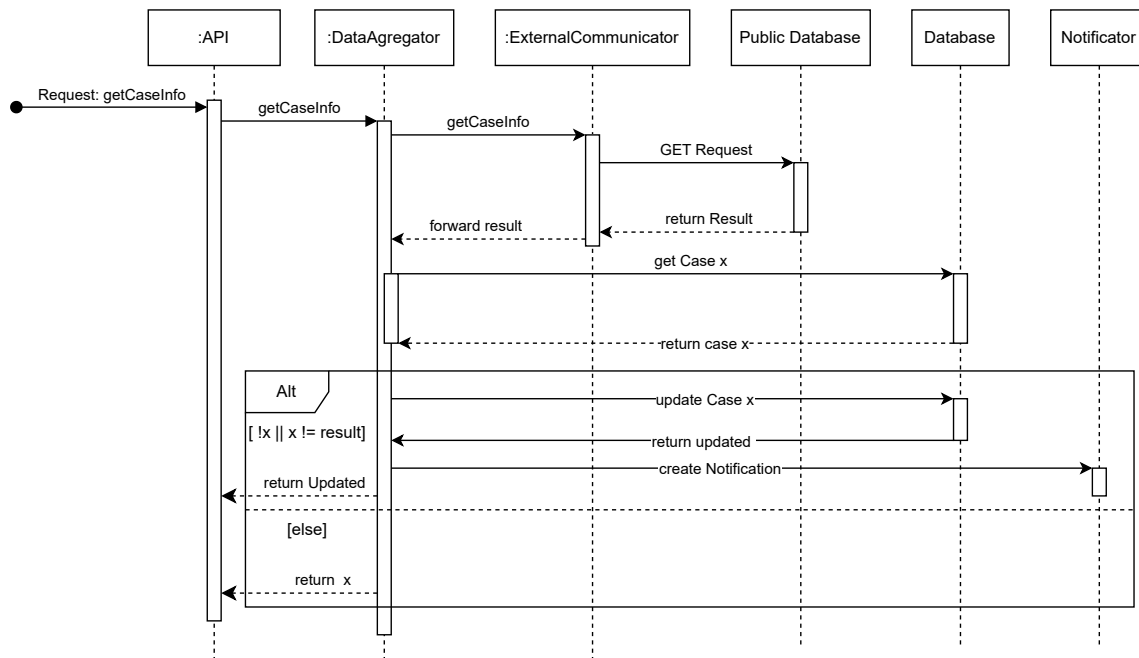
4.2.2 Informace o insolvenčním řízení

Insolvenční rejstřík je důležitým zdrojem informací pro ověřování probíhajících insolvenčních řízení vztahujících se k osobám, získání historie insolvenčních řízení a sledování změn v těchto řízeních. Systém by měl být schopen pravidelně kontrolovat aktuálnost těchto informací a upozorňovat odpovídající zodpovědnou osobu na případné změny, které by mohly mít vliv na právní zastoupení klienta.

4.2.3 Informace o soudních řízeních a jednáních

Pro získání informací o soudních řízeních a jednáních je možné využít databázi **infoSoud** a **infoJednání**, které byly blíže popsány v části 2.2. Tyto informace jsou důležité pro právníky, jelikož musí pravidelně sledovat průběh soudních řízení, účastnit se soudních jednání v zájmu svých klientů a v souvislosti s tím plánovat svou práci. Systém tak bude automaticky získávat tyto informace z obou databází a pravidelně je aktualizovat, aby byly uživatelům vždy k dispozici nejnovější údaje. Kromě toho bude systém informovat zodpovědné osoby

o důležitých změnách v soudních řízeních a jednáních, aby mohly adekvátně reagovat na vývoj situace a přizpůsobit svou práci podle potřeb klientů. Následující sekvenční diagram přibližuje proces a využití jednotlivých modulů pro získání dat z těchto databází, ověření změn a případné uložení získaných dat a vytvoření notifikace v případě změn. Proces by mohl být obdobný pro **insolvenční rejstřík**.



Obrázek 4.4: Sekvenční diagram znázorňující proces získání dat z externích systémů s případnou aktualizací dat.

4.3 Datový model a návrh databáze

Tato podkapitola bude věnována analýze datových entit vystupujících v systému a následnému návrhu databáze pro vytvářený systém. Nejprve je provedena identifikace a popis datových entit včetně jejich atributů. To umožní stanovit, jaká data bude třeba pro účely systému ukládat. Poté je na základě těchto znalostí vytvořen Entity-Relationship (ER) diagram, který pomůže vizuálně zobrazit a lépe pochopit vztahy mezi jednotlivými datovými entitami. Bude dále sloužit jako podklad pro vytvoření schématu databáze, která bude z ER diagramu přímo vycházet. Pomůže s efektivním návrhem databáze a pochopením celkového datového modelu. Každá entita bude samozřejmě obsahovat i unikátní identifikátory, které nejsou ve výpise atributů uváděny.

4.3.1 Identifikace a popis datových entit

V podkapitole jsou popsány datové entity a jejich atributy a je popsán vztah mezi těmito entitami. Tyto informace budou následně použity pro návrh ER diagramu, který znázorní vztahy mezi datovými entitami vizuálně. U každé entity je uveden její krátký popis a důvod zahrnutí do systém následovaný popisem vybraných atributů.

Advokátní kancelář

System by měl být využitelný několika nezávislými právními kanceláři. Je tak potřeba ukládat informace o každé z nich. Jedná se o něco jako centrální entitu, na kterou budou jistým způsobem navázány všechny ostatní entity. Jedna advokátní kancelář v podstatě využívá jednu instanci systému - veškeré ostatní entity jsou od sebe oddělené na základě kanceláře.

Zaměstnanci

Vystupují též jako primární aktéři systému a budou jeho uživateli. Je proto nezbytné ukládat informace o nich. Jsou přímo navázány na advokátní kancelář, ve které pracují. Jak bylo znázorněno v 4.1.2, mají také přiřazené role v rámci advokátní kanceláře a s nimi související pravomoci.

Klienti

Advokátní kanceláře poskytují své služby několika klientům a informace o každém z nich je tak potřeba uchovávat. Jedná se o jejich osobní data a další informace související s řešením případů s nimi jako vlastní poznámky či stav jednání a další.

Spis

Spisy reprezentují vždy konkrétní případ, který je řešen s klientem. Jsou tak vždy navázány na klienta, který může mít více spisů nebo také žádný. Má vlastní název, libovolný popis, stav ve kterém se řešený případ nachází a může být také propojený se zodpovědným zaměstnancem.

Spisová značka

Spisová značka, přidělena soudem k případu. Ke spisu v systému jich může být přiděleno více. Obsahuje unikátní identifikátor sestavený z parametrů popsanych v 2.2, a je navázána na soudní řízení.

Subjekt

Subjekt reprezentuje jakoukoli další entitu figurující v rámci spisu, která není klientem. Subjekt tak může být protistrana, nebo např. svědek či jiná zúčastněná osoba nebo instituce a jsou tak potřeba ukládat její osobní údaje.

Kontaktní osoba

Ke klientovi, protistraně a dalším subjektům ve spisu může náležet několik kontaktních osob, které k němu mají nějaký vztah. U právnických osob to mohou být různí zaměstnanci se kterými je jednáno, u fyzických osob například příbuzní, zastupující osoby a podobně, se kterými je v rámci případu také komunikováno. Atributy by měly být volitelné, jelikož nemusí být vždy důležité.

Adresář

System má umožnit ukládání a organizaci dokumentů, je proto nutné v podstatě simulovat souborový systém se složkami. Advokátní kancelář tedy bude mít jeden svůj kořenový adresář, do kterého mohou být přidávány další složky a soubory. Každý adresář tedy může obsahovat několik dalších adresářů a souborů a kromě kořenového má vždy jeden nadřazený adresář. Každý klient a každý spis má svůj “kořenový” adresář.

Soubor

Entita reprezentující konkrétní soubor či dokument v rámci souborového systému. Soubor vždy patří do nějaké složky. Jelikož je vyžadované i uchování více verzí souborů, musí tato informace tak být také zahrnuta.

Aktivita

Jedná se o nějakou činnost či akci provedenou v rámci advokátní kanceláře, klienta nebo konkrétního spisu. Na úrovni spisu se může jednat například o konkrétní provedený právní úkon, jako sepsání smlouvy, na úrovni klienta může jít o obecný telefonát bez vazby na případ. Aktivita tak musí být navázaná buď na advokátní kancelář, klienta, nebo spis, přičemž všechny tyto entity mohou mít pochopitelně několik aktivit. Zaznamenané aktivity budou sloužit pro přehled dění v rámci advokátní kanceláře. Aktivita může být též naplánovaná do budoucna, jako například schůzka či lhůta. Aktivitu může vytvořit ručně zaměstnanec nebo samotný systém v rámci aktualizace dat a upozornění. Atributy aktivity se budou lišit na základě typu aktivity a nějaké informace tak mohou být pro jednotlivé typy vynechány. Mezi důležité údaje ukládané v rámci aktivity patří název a popis a čas vzniku či naplánovaný čas.

Úkon

System má umožňovat podporu pro účtování provedených služeb a proto je nutné zaznamenávat jednotlivé úkony provedené v rámci řešení případu klienta. Úkon tak bude reprezentovat každou provedenou činnost zaměstnancem v rámci jeho náplně práce. Zároveň tak úkony mohou sloužit pro celkový přehled vykonané práce zaměstnanci. Mezi důležité atributy bude patřit opět název úkonu, popis, čas či cena. Úkon může být zároveň navázán na spis, klienta, či samotnou advokátní kancelář.

Faktura

Faktura se vždy váže k nějakému klientovi a skládá se z několika položek, které k faktuře mohou být přidružené. Položky mohou být vytvořeny na základě úkonů a být s nimi tak propojené.

Pošta

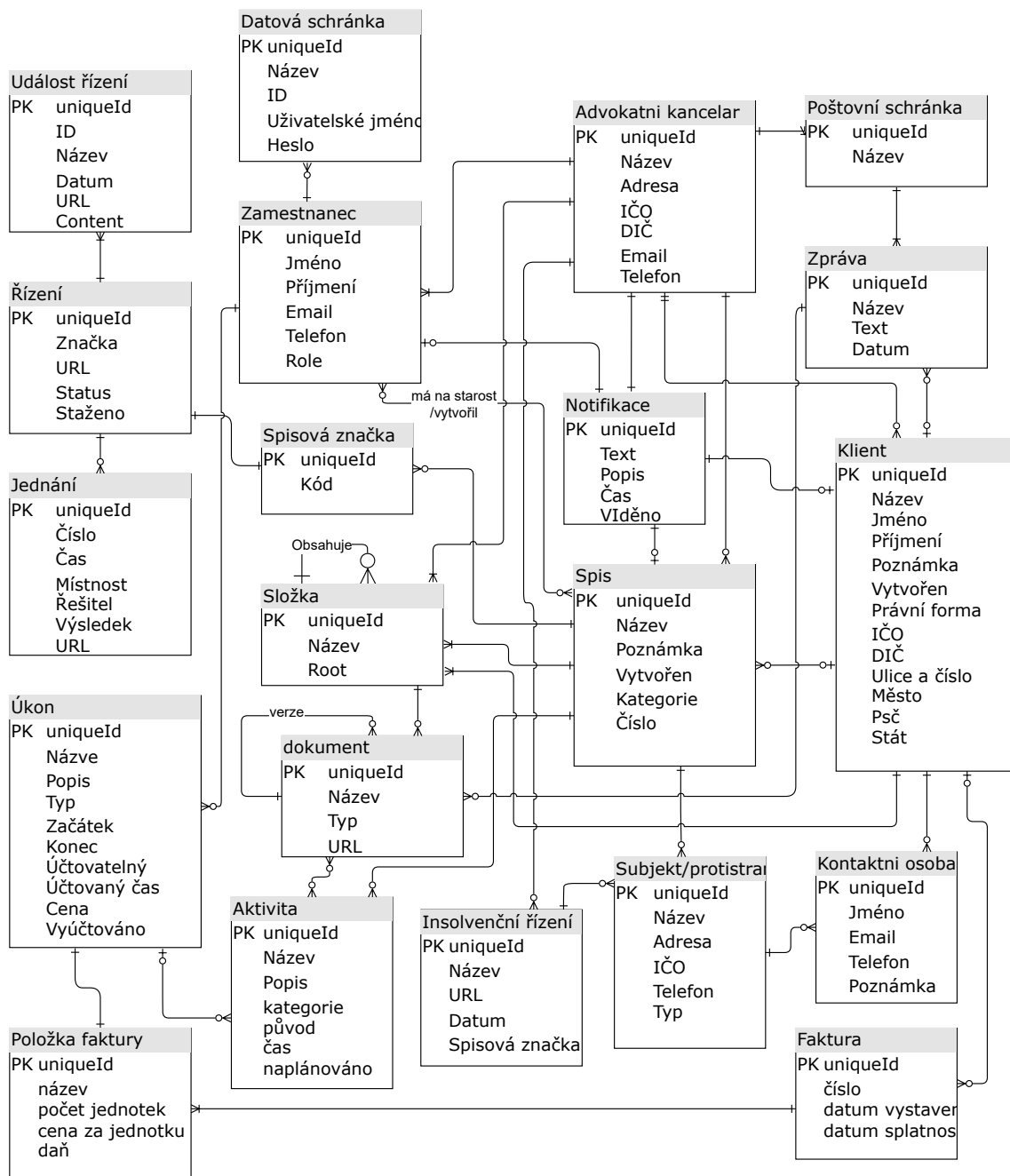
System musí umožňovat správu pošty a je potřeba tak ukládat jednotlivé zprávy a dopisy. Mezi uchovávané informace patří datum a čas přijetí, odesílatel, název a text i případná vazba na dokument v případě přiloženého souboru.

Entity spojené s agregovanými daty

Jedná se o ukládaná data z externích zdrojů popsaná v předešlé kapitole, kde je uveden již dostatek detailů. Patří sem entity: **Řízení, Událost řízení a jednání**. Tyto entity jsou přímo navázané na spis či klienta. Data je potřeba ukládat do databáze, aby bylo možné kontrolovat případné nastalé změny a informovat o nich uživatele.

4.3.2 ER diagram

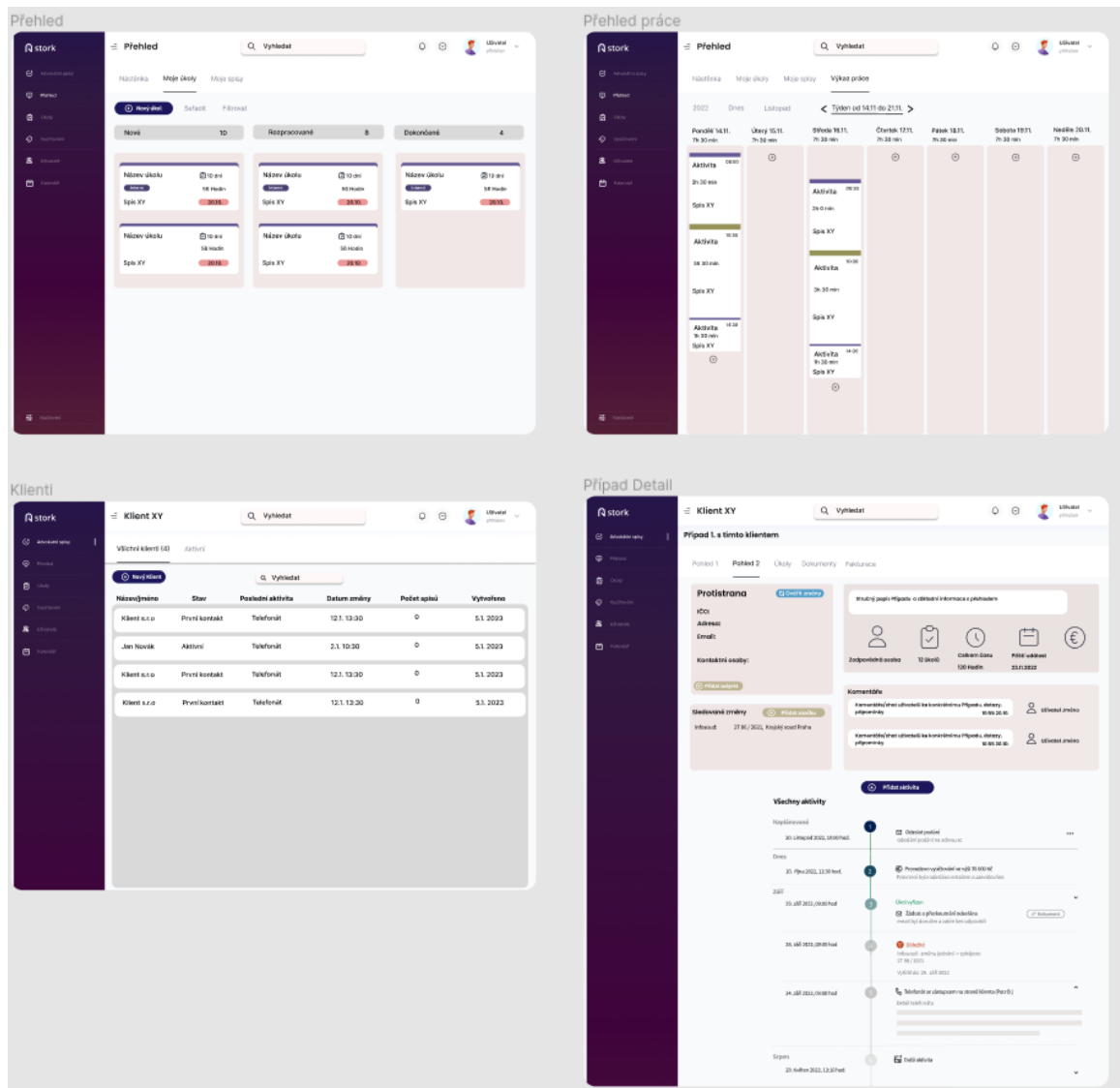
V této podkapitole se zaměříme na vizualizaci vztahů mezi datovými entitami popsanými v předešlé kapitole systémem prostřednictvím ER diagramu. Tento diagram slouží jako grafický nástroj používaný pro návrh databází, pomocí kterého je možné znázorňovat jednotlivé entity, jejich atributy a vztahy mezi nimi, což umožňuje lepší pochopení celkové struktury a organizace dat v systému.



Obrázek 4.5: ER diagram znázorňující datové entity figuruující v systému včetně jejich vybraných atributů a vazeb mezi nimi.

4.4 Funkcionalita a uživatelské rozhraní

V následující části jsou již blíže popsány vybrané funkce vyžadované od systému. Toto je vždy navázáno návrhem uživatelského rozhraní, které bude dané funkce zpřístupňovat uživateli. Cílem je navrhnout uživatelsky přívětivé rozhraní, díky kterému bude systém jednoduché používat. Rozhraní by mělo být zároveň graficky přitažlivé, jelikož to je jeden z hlavních aspektů, na základě kterého si uživatel udělá o systému první dojem, při jeho vybírání. Pro návrh vzhledu uživatelského rozhraní a jeho jednotlivých komponent bylo využito nástroje Figma¹. Ten umožňuje rychlé prototypování, znázornění návaznosti obrazovek a efektivní znovupoužití již navržených komponent.



Obrázek 4.6: Pohled na návrh uživatelského rozhraní v nástroji Figma s návazností jednotlivých obrazovek.

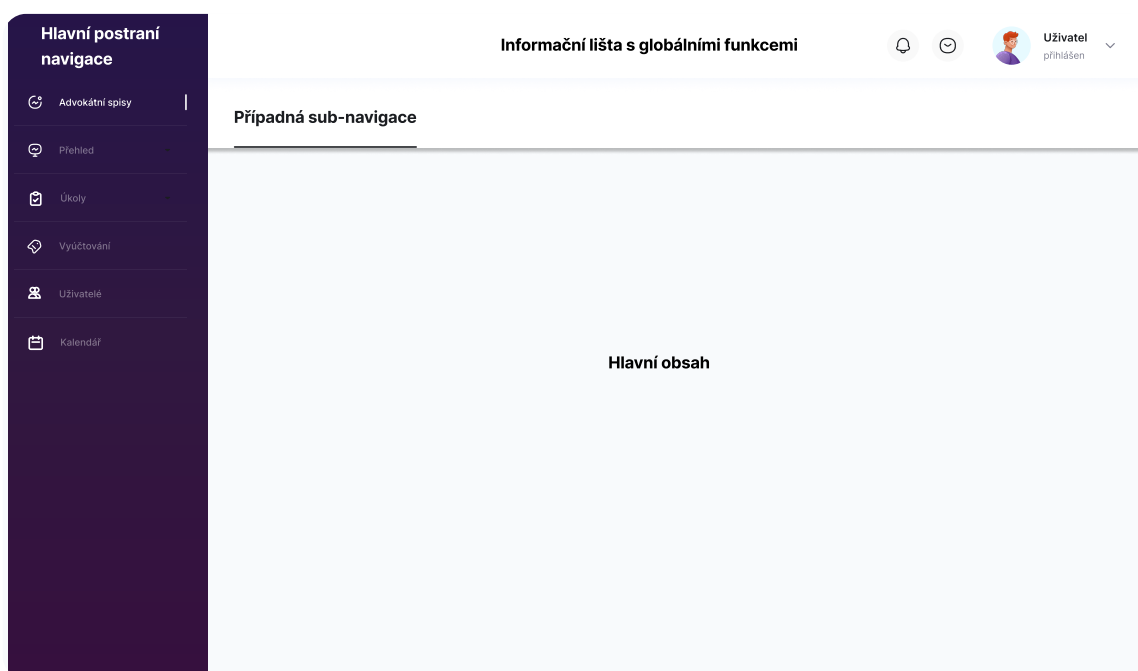
¹<https://www.figma.com/>

4.4.1 Celkové rozložení rozhraní

Systém bude disponovat rozložením znázorněným na následujícím obrázku. Pomocí hlavní **postranní navigační lišty** systému budou přístupné hlavní funkce či jinak řečeno jednotlivé hlavní části systému. Výhody hlavní postranní navigace spočívají v možnosti viditelného rozvinutí jednotlivých položek menu a možnosti jednotného zobrazení na všech velikostech displeje - na mobilních zařízeních může být pouze celé skryto a následně zobrazeno uživatelem v případě potřeby.

Informační lišta v horní části bude sloužit k zobrazení globálně dostupných funkcí, jako jsou upozornění, či správa uživatelského účtu. Pod ní je možné umístit lištu pro případnou **sub-navigaci**, která bude potřebná pro zpřístupnění některých funkcionalit přístupných v jednotlivých hlavních částech systému, jako je zobrazení klienta či spisu.

Zbývající plocha bude využita pro zobrazení hlavního obsahu a s ním spojených funkcí.



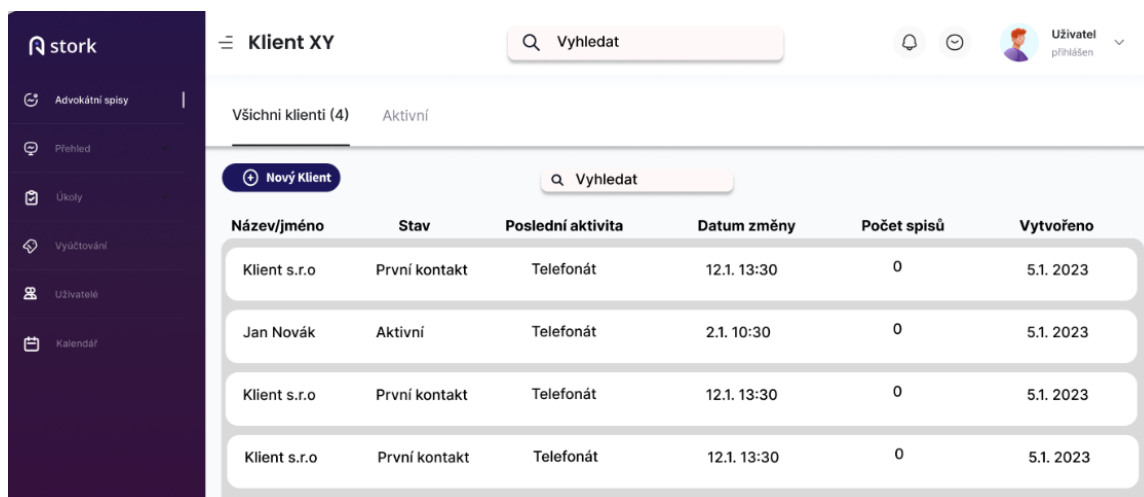
Obrázek 4.7: Návrh celkového základního rozložení uživatelského rozhraní v nástroji s postranní a horní navigací.

4.4.2 Hlavní přehled

Úvodní stránka systému, sloužící v podstatě jako **domovská obrazovka** uživatele, na které skončí vždy když se přihlásí do systému. Měla by sloužit jako místo, kde se uživatel dozví přehled všeho **nejdůležitějšího**, co se v nedávné době dělo v rámci advokátní kanceláře ve spojení s jeho hlavní agendou, jaké změny nastaly v rámci klientů a jejich spisů, či jaké události nebo schůzky má naplánované. Tento účel by mohla dobře splnit **časová osa aktivit**, která by zobrazovala chronologicky seřazené agregované aktivity nastalé či provedené v souvislosti se všemi klienty a všemi jejich spisy. Uživatel by tak získal okamžitý přehled o nedávném a aktuálním dění. Na hlavní stránce je dále umístěn panel se spisy přiřazené přihlášenému uživateli, který urychlí přístup k nim. Zobrazen je také panel činností daného uživatele.

4.4.3 Přehled klientů a spisů

Aby advokátní kanceláře měly dobrý přehled o svých klientech, je možné zobrazit si **všechny klienty** včetně těch potenciálních na jedné stránce se všemi důležitými atributy, které se vztahují ke každému z nich. Užitečné by mohly být funkce pro vyhledávání či řazení klientů podle jednotlivých sloupců. Vhodné by mohlo být také vytvoření více náhledů na základě stavu, který patří mezi hlavní parametry.



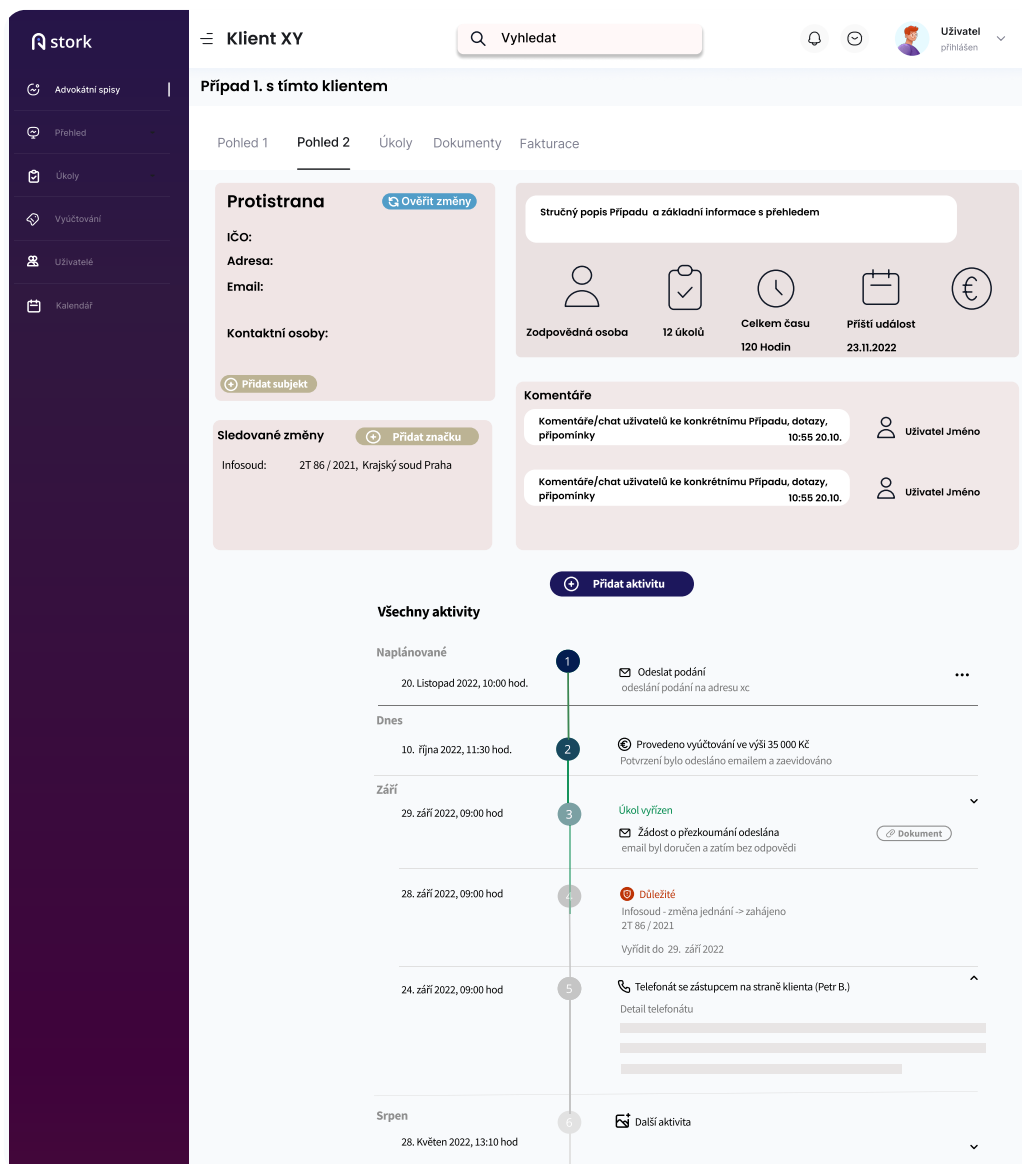
Obrázek 4.8: Návrh uživatelského rozhraní pro stránku s přehledem všech klientů.

4.4.4 Detail klienta a spisu

Při vybraní klienta je zobrazen jeho detail na **samostatné stránce**. Zde je potřeba mít přehled o všem souvisejícím s daným klientem. V první řadě jsou zde zobrazeny detailní informace o klientovi a jeho případných **kontaktních osobách** s možností úprav. Dále se jedná o **zodpovědné osoby** z advokátní kanceláře mající klienta na starost, přehled související s případy klienta a vyúčtováním a v neposlední řadě opět **časová osa**, která poskytuje přehled nejdůležitějších předešlých aktivit/činností souvisejících s klientem spolu s agregovanými aktivitami ze všech případů s ním řešených. Mezi funkcionalitu přístupnou z této stránky patří kromě úpravy klienta též možnost vyvolání automatické kontroly údajů z rejstříků, přidávání aktivit, poznámek a navigace na další entity související s klientem.

Jelikož se ke klientovi váže více hlavních entit jako jsou případy, dokumenty, provedená práce a vyúčtování, jsou tyto přístupné přes horní sub-navigaci s možností zobrazit je na celé stránce a zároveň zachováním znázornění jasné souvislosti s konkrétním klientem. Zobrazení pro detail spisu vypadá velmi obdobně, s tím že jsou zobrazeny jen detailní informace o daném spisu, časová osa aktivit souvisejících s daným spisem a stejně tak dokumenty vázající se ke spisu spolu s vykázanou prací ve spisu dostupné též skrz horní sub-navigaci.

Detail spisu již neobsahuje kartu pro vyúčtování, jelikož to je vázáno jen ke klientovi samotnému. Detail spisu navíc disponuje možností přidáním jedné či více **spisových značek** případu přidělených soudem, pokud je případ součástí probíhajícího soudního řízení. Přidání spisové značky umožní získat a agregovat veškeré informace o soudním řízení a jeho událostech a potenciálních změnách na základě této spisové značky. Spis typicky reprezentuje nějaký případ řešený s klientem, ve kterém mohou vystupovat **další subjekty**, a je zde tedy také možnost tyto subjekty přidávat.



Obrázek 4.9: Návrh uživatelského rozhraní pro stránku detailu spisu.

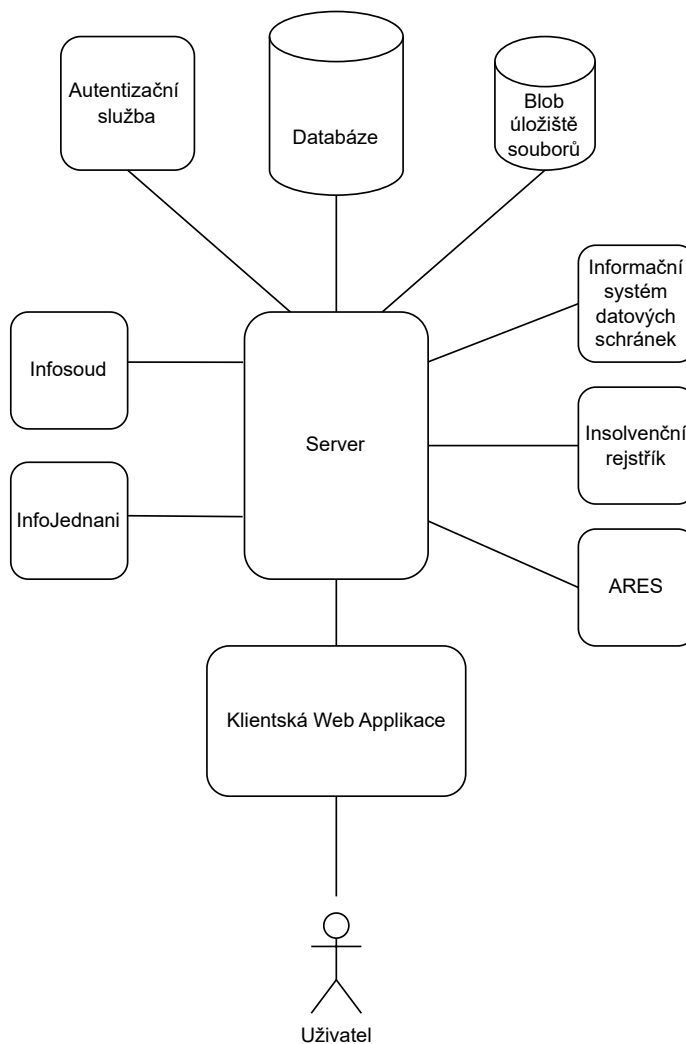
4.5 Návrh architektury systému

V této podkapitole je popsán návrh celkové architektury systému a je přiblížena struktura jednotlivých vrstev architektury. Bude se jednat primárně o třívrstvou architekturu, jelikož je potřeba umožnit přístup k systému několika uživatelům zároveň, je potřeba zpracovávat data na základě jejich akcí a data ukládat, k čemuž bude potřeba databáze. Bude však částečně využito i mikroservices přístupu, kde například pro část autentizace bude použita externí služba.

4.5.1 Vysokoúrovňové schéma systému

Toto schéma poskytuje základní přehled o propojení hlavních jednotlivých částí systému. Uživatel interaguje s klientem, což bude v případě tohoto systému webová aplikace běžící

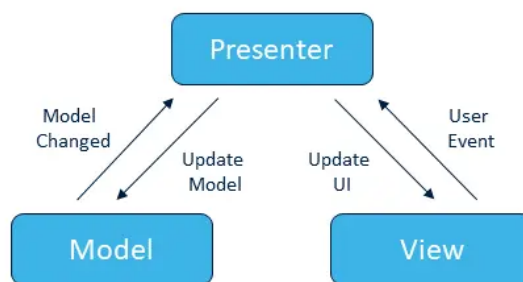
v prohlížeči na libovolném zařízení, která zajišťuje přístup k funkcím systému. Klient komunikuje se serverem, který zpracovává požadavky od klienta a vrací požadovaná data. Server je propojený s databází a je zodpovědný za provádění operací jako načítání, aktualizace a ukládání dat. Server také komunikuje s externími systémy, mezi které patří veřejné databáze a Informační systém datových schránek. Dále je server napojený na autentizační službu pro autentizaci uživatelů a cloudové úložiště souborů pro ukládání dokumentů.



Obrázek 4.10: Vysokoúrovňové schéma systému, které nabízí pohled na systém jako celek a znázorňuje propojení s externími systémy.

4.5.2 Architektonický vzor MVP

Architekturu implementovaného systému by bylo možné popsat pomocí architektonické vzoru MVP (Model-View-Presenter), ten je podobný známému vzoru MVC avšak rozdíl je v tom, že vrstva View (frontend s uživatelským rozhraním) nekomunikuje přímo s Modelem (datovou vrstvou), ale veškerá komunikace probíhá skrze Presenter, který jak manipuluje Model na základě akcí z View, tak aktualizuje View na základě dat získaných z modelu.[48]



Obrázek 4.11: Schéma architektonického vzoru Model-View-Presenter. Převzato z: [\[https://support.touchgfx.com/4.17/docs/development/ui-development/software-architecture/model-view-presenter-design-pattern\]](https://support.touchgfx.com/4.17/docs/development/ui-development/software-architecture/model-view-presenter-design-pattern)

4.5.3 Schéma architektury systému

Toto schéma k vidění na obrázku 4.12. odráží architektonický vzor MVP a nabízí již bližší pohled na konkrétní prvky v architektuře systému. Znázorňuje hlavní části v jednotlivých vrstvách architektury, které budou potřeba pro zajištění vyžadované funkcionality a umožňuje pohled na závislosti mezi těmito částmi systému

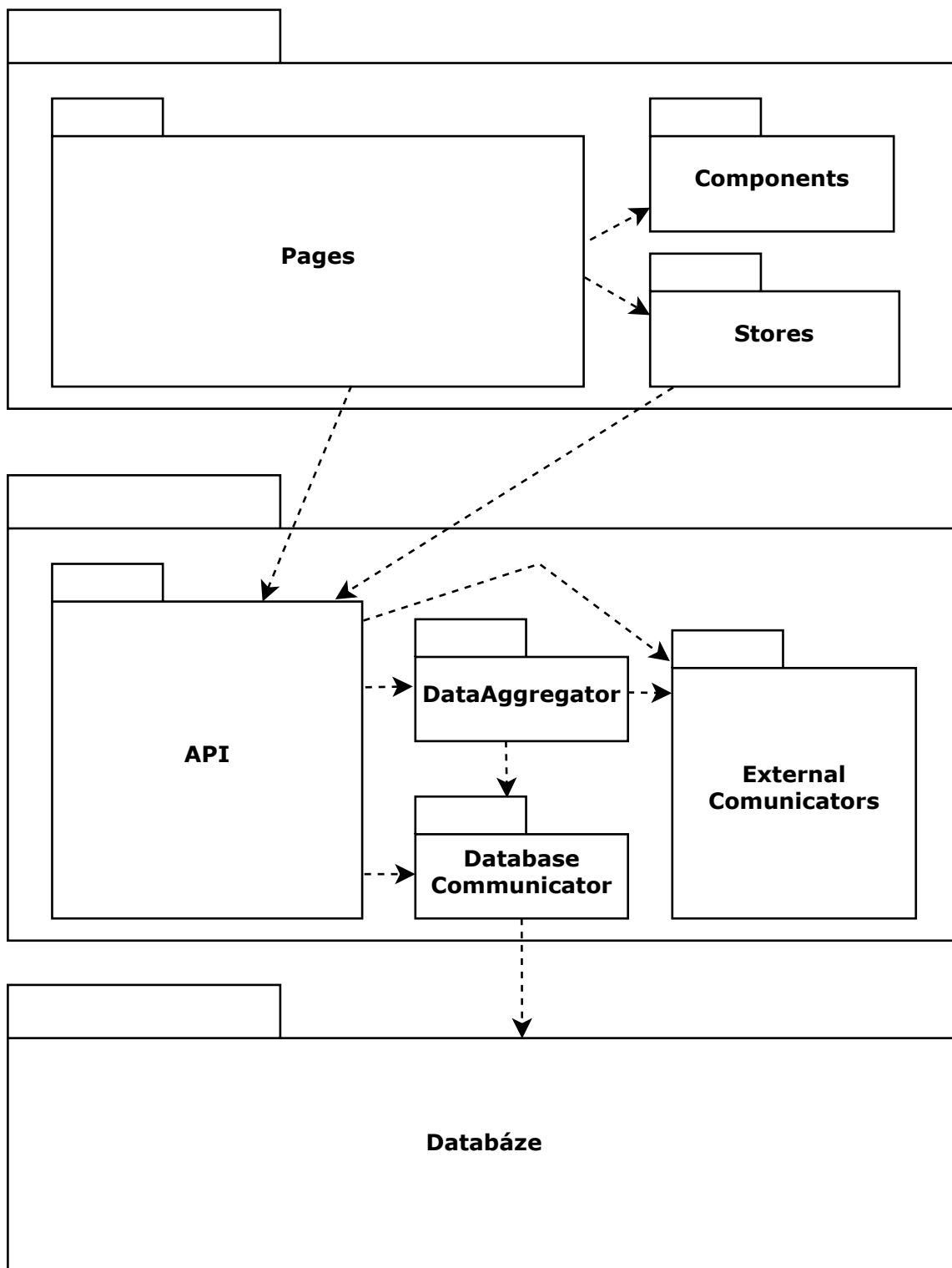
Vrstva View bude zajišťovat uživatelské rozhraní, pomocí kterého uživatelé budou interagovat se systémem. V této vrstvě tak půjde o zobrazování dat uživatelům, reagování na akce uživatelů a jejich následné předávání do Presenteru. Bude se skládat z následujících hlavních částí:

- **Pages:** Bude obsahovat implementaci pro jednotlivé obrazovky uživatelského rozhraní.
- **Stores:** Funkční celky, které budou zajišťovat správu stavu uživatelského rozhraní a jednotlivých komponent.
- **Composables:** Soubory funkcí, které umožní sdílení logiku mezi různými komponentami.

Presenter je jádrem MVP architektury a v tomto případě půjde o server. Jeho úkolem bude zpracovávat akce z View, manipulovat s Modelem a aktualizovat View na základě dat z Modelu.

- **API:** Bude obsahovat rozhraní pro komunikaci s View .
- **Data Aggregator:** Bude se starat o shromažďování a zpracování dat z různých externích zdrojů, sjednocovat jejich formát a ukládat data do databáze.
- **External Communicators:** V této části budou implementovány jednotlivé moduly pro komunikace s externími systémy a službami.
- **Database Communicator:** Modul, který se stará o komunikaci s databází.

Model je v rámci systému databáze, kde budou uchována data nutná pro fungování systému a bude obsahovat tabulky na základě popsanych datových entit v kapitole 4.3.1.



Obrázek 4.12: Schéma architektury systému, které nabízí bližší pohled na strukturu jednotlivých vrstev.

Kapitola 5

Implementace systému pro právní kanceláře

Po zpracování návrhu systému bylo cílem práce navržený systém implementovat. Bylo k tomu využito několika hlavních implementačních technologií, blíže popsanych již v teoretické části v kapitole 3.3 a s pomocí dalších volně dostupných knihoven pro usnadnění řešení často opakujících se implementačních problémů. Bylo také využito externí služby pro autorizaci uživatelů pro zjednodušení implementace této části systému a také zvýšení úrovně zabezpečení. V této části je blíže popsána celková koncepce systému z pohledu implementace a je znázorněna struktura celého projektu. Dále jsou zde více do detailu popsány implementace vybraných částí systému a je znázorněna celková architektura s náhledem na všechny funkční celky.

5.1 Struktura projektu

Na základě požadavků na systém a jeho návrhu, bylo rozhodnuto pro implementování systému formou **webové aplikace**. To umožní jednoduchou distribuci systému mezi uživatele, bez nutnosti instalace na koncová zařízení. Bude možné nabídnout **multiplatformní dostupnost**, díky možnosti přístupu z libovolného **webového prohlížeče** a případné změny či aktualizace systému je možné provést přímo zveřejněním nové verze na hostitelský server, bez nutnosti zásahu uživatele. Po tomto rozhodnutí byly následně na základě zvážení více alternativ, vlastních preferencí a předešlých zkušeností zvoleny již zmíněné implementační technologie. V této části je blíže přiblížena již konkrétní struktura projektu celého implementovaného řešení.

Rozdělení závislostí a oddělení funkcionality

V rámci respektování dobrých principů tvorby software, systém je tvořen modulárně, se snahou o zachování samostatné zodpovědnosti pro každý modul a dodržování jejich vzájemného oddělení. [54] Souborová struktura projektu pro přehlednost do jisté míry reflektuje celkovou navrženou architekturu systému, tak jak byla znázorněna schématem 4.12. Zjednodušeně se je možné na strukturu projektu dívat jako na Frontend, Backend a Databázi.

Na straně frontendu jsou nejzásadnějším adresářem **Pages**, kde jsou implementované hlavní **pohledy aplikace**. Z pohledu webové aplikace reprezentují jednotlivé stránky, které jsou přístupné pod vlastní url adresou, které jsou díky frameworku nuxt vygenerovány pro všechny soubory v této složce. Framework nuxt navíc umožňuje pokročilejší konstrukce při organizaci adresářové struktury pro tvorbu automatické navigace a návaznost **jednotlivých stránek**. V adresáři *pages* je tak možné vytvářet další adresáře reprezentující “podstránky” se základní url danou názvem adresáře. Soubory a adresáře je možné takto libovolně vnořovat a větvit. Navíc je možné v názvu stránek nechat místo pro dynamické parametry, které umožňují dynamickou navigaci a získávání odpovídajících dat na základě dynamické url adresy. [20] Příkladem může být například získání klienta s určitým id a zanechání této informace přímo v adrese url třeba i spolu s jeho jménem.

Každý soubor v adresáři *pages* je soubor frontend frameworku *Vue* a vždy obsahuje část *template*, reprezentující vzhled dané stránky a je tvořen kombinací *HTML* prvků, **vlastních komponent** a komponent z využití knihovny *vuetify*¹. K elementům v templatu se váží přidružená data, která mohou být získána dynamicky a funkce, které jsou definované v druhé části *vue* souborů. Část *script*, neboli funkční a datová část, obsahující importy externích komponentů a funkcí, definice reaktivních proměnných, funkce pro aktualizace a manipulaci zobrazovaných dat a další. [26] Bližší struktura adresáře *pages* je k vidění ve schématu architektury frontendu 5.1.

Úzce souvisejícím adresářem je adresář *Components*, který taktéž obsahuje soubory *vue*, tentokrát implementující menší znovupoužitelné komponenty², které zapouzdřují specifické prvky či celky uživatelského rozhraní s definovaným vzhledem a relevantními vlastními funkcemi. **Komponenty** jsou poté importovány a využívány v jiných větších komponentách nebo v hlavních stránkách. Je u nich využíváno principu předávání tzv. props z nadřazených komponentů, které umožňují předat komponentám variabilní data a blíže specifikovat jejich chování. [26]

5.1.1 Klientská část

Celý frontend aplikace se neustále nachází v nějakém **stavu**, který se dynamicky mění na základě akcí uživatele a zobrazovaných datech a jeho změny musí být reflektovány v uživatelském rozhraní. K tomuto jsou využívány tzv. **Stores**, které uchovávají globálně přístupný stav s definovanými reaktivními proměnnými a funkcemi pro jejich aktualizaci. Ke správě stavu v celé aplikaci bylo využito knihovny *Pinia*³. Adresář *Store* obsahuje implementaci pro 4 hlavní stavové objekty, kterých je v aplikaci využíváno a jsou také k vidění v následujícím schématu architektury 5.1. Jedná se o:

- **Global Store** udržující globální stav aplikace s daty jako informace o přihlášeném uživateli, advokátní kanceláři, klientech a spisech.
- **Activity Store** udržující stav o aktuálně načtených aktivitách pro AK, klienta či spis a obsahuje související funkce.
- **Work Store** má na starost stav související se zaznamenanými úkony uživatele, přehledem úkonů a vyúčtováním.

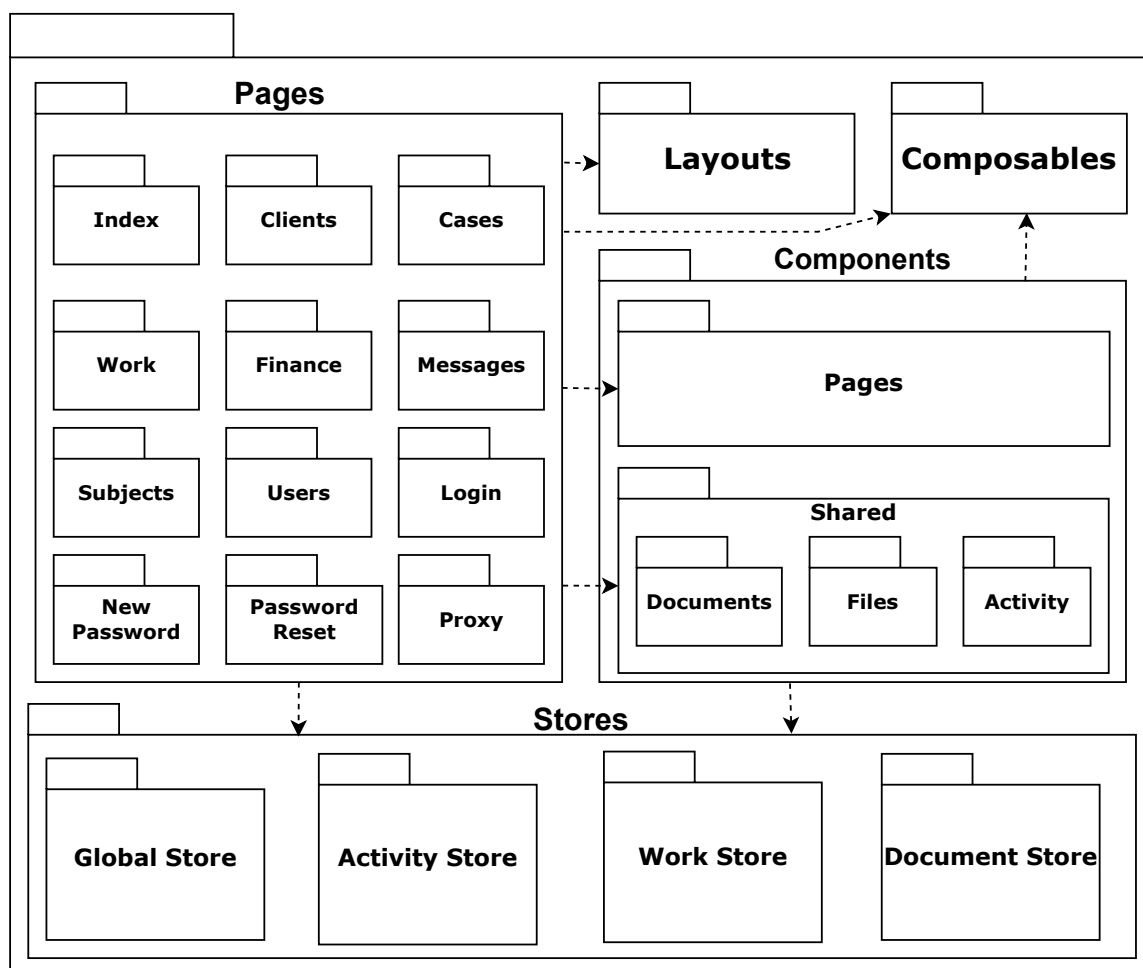
¹<https://next.vuetifyjs.com/>

²<https://vuejs.org/guide/essentials/component-basics.html>

³<https://pinia.vuejs.org/>

- **Document Store** souvisí s udržováním a aktualizací veškerých stavů souvisejících s prací s dokumenty a složkami.

Mezi další menší adresáře na straně frontendu patří ještě adresář **Layouts**, obsahující hlavní implementaci pro **rozložení systému**, neboli “obalující části”, která je shodná pro všechny stránky systému a patří sem tak navigační lišta, postranní navigace a implementace/volání některých funkcí, které je třeba vykonat globálně při prvním načtení systému. Dále adresář **composables**, obsahující implementace pro globální **pomocné funkce**, které je možné využít na straně klienta, ale i serveru. Patří sem například funkce pro ověřování pravomocí, zobrazování chybových hlášek, různá formátování, například datumů, konstanty a další. Adresář plugins obsahuje definice a zakomponování některých externích knihoven do projektu. Assets obsahuje globální definice kaskádových stylů, obrázky a soubory s překlady do dalších jazyků. V současné době jsou připraveny soubory pro český jazyk, anglický jazyk a německý jazyk, kde kompletní je jen ten pro český jazyk. Pro překlady je využíváno knihovny vueI18n⁴.



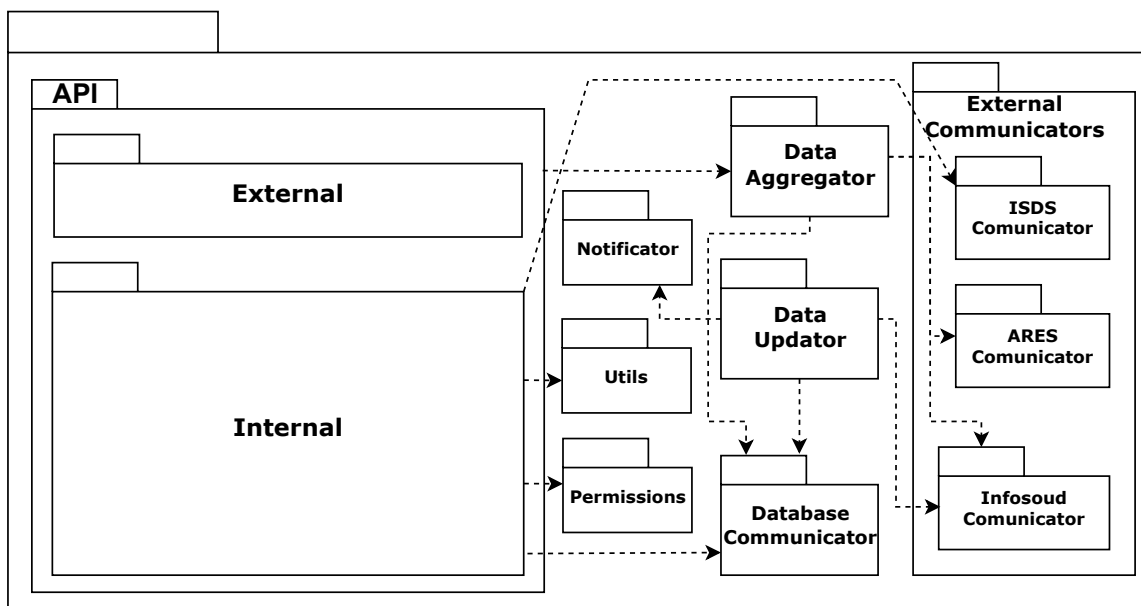
Obrázek 5.1: Detailní náhled architektury frontendové části aplikace (View)

⁴<https://vue-i18n.intlify.dev/>

5.1.2 Serverová část

Veškerá serverová část implementace, neboli backend systému se nachází v adresáři Server, který je rozdělen do 2 podadresářů API a Backend. V Adresáři API jsou definované veškeré koncové body REST API, které jsou volané jak z frontendu aplikace, tak některé z nich i zvenčí systému. Zde je řeč hlavně o zpřístupnění funkcí pro agregaci dat z veřejných zdrojů. Pro jejich snadnou dostupnost a možnou budoucí využitelnost třetími stranami jsou v adresáři external implementovány koncové body API, které budou po zveřejnění aplikace veřejně přístupné a umožní jednoduché získání dat v čitelném formátu z nějakých z vybraných zdrojů popsaných v 2.1. Pro zamezení přetížení systému a případného nevhodného využití by však mohlo být do budoucna žádoucí implementovat základní způsob ověření uživatele a umožnit přístupnost funkcí jen s platným přiděleným API tokenem. Zbývající API implementace využívaná aplikací je především na získávání požadovaných dat z databáze na základě query parametrů, a jejich transformace do očekávaného formátu, či vytváření a aktualizace hodnot v databázi.

Adresář backend obsahuje ostatní serverové moduly, které nejsou přímo dostupné jako API koncové body. Patří sem primárně důležité moduly pro získávání informací z veřejných databází, zpracování a agregaci těchto externě získaných dat a moduly pro jejich ukládání a pravidelnou aktualizaci. Tyto moduly jsou detailně popsány v nadcházející části 5.3. Dále funkce pro kontrolu pravomocí, notifikační modul, modul notifikací, modul pro komunikaci s databází pro specifické operace a další pomocné funkce na straně serveru v souboru utils.js například pro šifrování a dešifrování hesel ukládaných v databázi.



Obrázek 5.2: Detailní pohled na architekturu backendu systému (presenter)

5.1.3 Databáze

Implementace databáze by se dala zahrnout do backendové části systému, kde jsou implementovány veškeré funkce pro komunikaci s databází. Samotná definice schématu se v projektu ale nachází samostatně v adresáři Prisma vzhledem k běžnému postupu při využívání této knihovny a následováním oficiální dokumentace pro její využití spolu se zbývajícími zvo-

lenými technologiemi. Definice veškerých tabulek databáze je provedena skrz ORM Prisma pomocí definice modelů, řídicích se principy popsaných v dokumentaci. Umístěny jsou v souboru `schema.prisma`. Definované modely a vazby mezi nimi přímo reflektují navrženou databázi znázorněnou ER diagramem v příloženém schématu 4.3.2.

5.2 Uživatelské účty a zabezpečení

Přihlašování a správa uživatelů včetně správy jejich rolí tvoří základ každého informačního systému a tento případ není výjimkou. V této kapitole je tak popsán konkrétní způsob jak je tato funkcionální implementována v rámci tohoto systému. Jak bylo nastíněno v návrhu, systém musí umožnit přihlašování zaměstnanců advokátní kanceláře do systému a to pod různými jim předělenými rolemi, které jsou blíže popsány v 4.1.2. Odpovídajícím způsobem pak musí být také zajištěna autorizace přístupu k jednotlivým funkcím systému a datům na základě definování vedoucím dané advokátní kanceláře. Každý uživatel s rolí vedoucí má možnost přidat nové uživatele do systému a definovat jejich role, ze kterých bude vyplývat, k jakým funkcím systému a k jakým datům v systému budou mít tito uživatelé přístup.

5.2.1 Správa účtů a autentizace uživatelů

Jelikož problematika autentizace uživatelů a vytváření uživatelských účtů je v dnešní době velmi rozšířená, existuje již velké množství volně dostupných řešení, které toto efektivně umožňují a je možné je jednoduše zakomponovat do vlastního projektu. Je tak výhodné těchto řešení využít a to i z důvodu větší bezpečnosti. Po zvážení více alternativ jsem se rozhodl pro využití služby Supabase⁵, která nabízí přívětivé api rozhraní pro registraci a autentizaci uživatelů, včetně možností resetu hesel a dalších souvisejících funkcí. Služba je pro jednoho uživatele zdarma pro první dva projekty a současně nabízí zdarma hosting postgres databáze propojené s projektem, což bude pro vytvářený systém potřeba také. Bude tak možné spojit využití jednoho poskytovatele pro obě vyžadované služby, což bylo také jedním z rozhodujících faktorů pro volbu Supabase. Mezi další patřily kvalitní dokumentace, kompatibilita a podpora pro ostatní zvolené technologie, především pro nuxt3. [25] Mezi zvažované alternativy patřily služby Auth0⁶ a Firebase⁷.

Mezi hlavní implementovanou funkcionálnítu spojenou s autentizací patří následující případy užití:

1. **Pozvání nového uživatele** - Zde se jedná o přidání nového zaměstnance do systému a vytvoření jeho nového uživatelského účtu vedoucím zaměstnancem. Z důvodu bezpečnosti a integrity osobních přihlašovacích údajů je uživateli poslána pozvánka k připojení do systému na email. Uživatel má následně možnost pozvánku do 24 hodin přijmout, nastavit si vlastní heslo pro přístup do systému a úspěšně tak dokončit vytvoření svého účtu. V okamžiku vytvoření pozvánky je využito funkce z knihovny Supabase `inviteUserByEmail`. V autorizační databázi Supabase je vytvořen nový uživatel a jeho data jsou vrácena včetně unikátní `userSupabaseId` identifikátoru. Poté jsou zadaná data nového uživatele uložena do vlastní postgres databáze spolu s nově vytvořeným `userSupabaseId`.

⁵<https://supabase.com/>

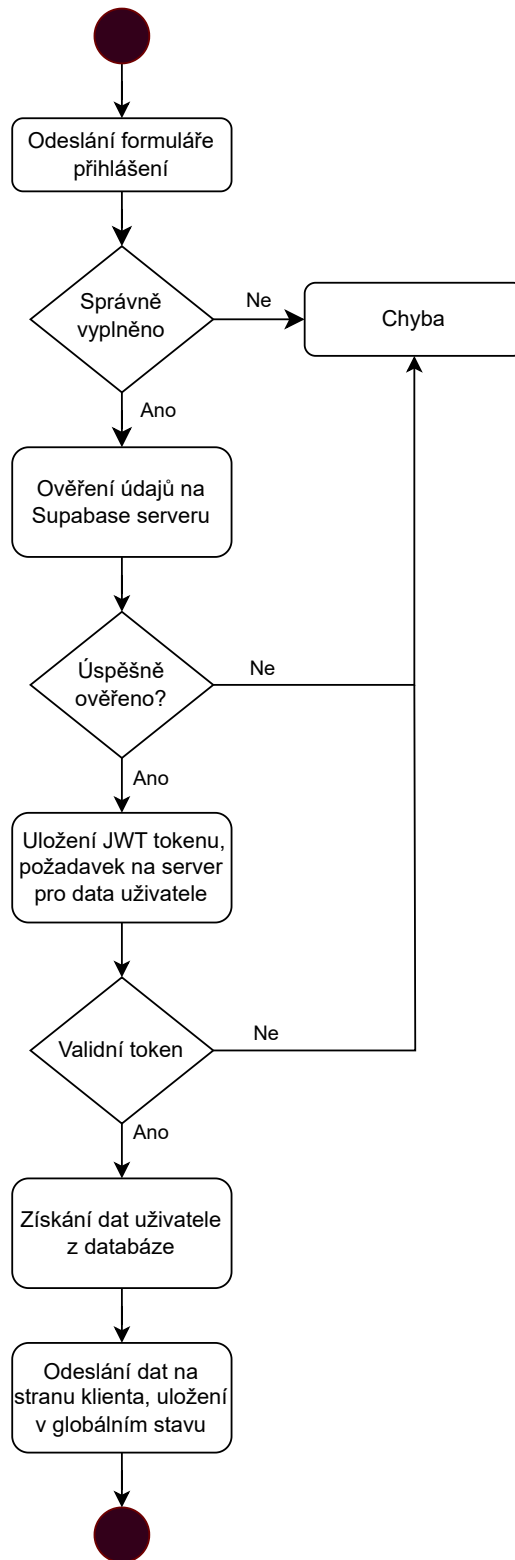
⁶<https://auth0.com/>

⁷<https://firebase.google.com/>

2. **Přihlášení do systému a odhlášení** - Přihlášení do systému je provedeno na základě emailu a hesla. Existence uživatele a správné heslo je ověřeno odesláním požadavku na server Supabase, který při úspěšném ověření vrátí jako odpověď informace o uživateli včetně vygenerovaného autentizačního JWT tokenu. Klientská knihovna Supabase tuto odpověď zpracuje a uloží JWT token do lokálního úložiště a je následně přikládán do autorizační hlavičky následujících http požadavků. Supabase objekt uživatele je možné poté získat kdekoli v aplikaci zavoláním funkce *useSupabaseUser*, která ověří platnost jwt tokenu se Supabase serverem a vrátí informace o uživateli.

V navázání na získání uživatele ze Supabase byla vytvářena vlastní implementace pro získání veškerého kontextu uživatele, jako jsou například jeho datové schránky, spisy apod. z vlastní postgres databáze. Vlastní API endpoint */context* tak při zavolání získá veškerá data uživatele pomocí vlastní funkce *getUser* implementované v souboru *getContext*. Funkce nejprve ověří uživatele pomocí Supabase na straně serveru skrze funkci *serverSupabaseUser* dostupné z knihovny Supabase. Ta získá JWT token z autorizační hlavičky požadavku a ověří ho dotazem na Supabase autorizační server. Pokud je token platný, vrátí supabase objekt uživatele. Uživatel je tedy přihlášen a je možné získat jeho informace z databáze na základě emailu. Je tak následně proveden dotaz do databáze na získání zaměstnance na základě dané emailové adresy a tato data jsou poté vrácena ve formátu json na stranu klienta, kde jsou poté uložena do globálního stavu. V případě odhlášení jsou poté tato data vynulována, JWT token je zneplatněn a odebrán. a uživatel je přesměrován na přihlašovací stránku.

3. **Reset/změna hesla** - Změna hesla se řídí standardizovaným pracovním postupem doporučeným Supabase dokumentací, podle které je implementován. Uživatel si může požádat o obnovu hesla zadáním emailové adresy. Ta je poté odeslána v požadavku na stranu serveru Supabase pomocí knihovni funkce *resetPasswordForEmail*. Pokud uživatel s daným emailem existuje, Supabase mu odešle email s předem specifikovaným odkazem pro přesměrování a přiloženým autentizačním tokenem. Při kliknutí na odkaz je uživatel přesměrován zpět do aplikace na specifikovanou stránku *new-password*, spolu s přiloženým tokenem v url adrese. Před přistáním na této stránce je provedeno ověření platnosti přiloženého tokenu na Supabase serveru. Pokud je v pořádku, je již následně uživateli umožněno nastavit si nové heslo, které je uloženo.
4. **Odstranění účtu** - Jednoduchá operace, kterou může provést uživatel s rolí vedoucí. Nejprve se provede odstranění uživatele a jeho dat z vlastní databáze a potom ze Supabase autorizační databáze využitím knihovni funkce *deleteUser()* s parametrem *userSupabaseId*, který je pro každého uživatele uložen ve vlastní databázi. [25]



Obrázek 5.3: Diagram aktivit znázorňující proces přihlášení do systému a autentizace

5.2.2 Autorizace

Autorizace je další nevyhnutelnou součástí informačních systémů a je potřeba k tomu, aby byl uživatelům umožněn přístup jen k takovým zdrojům a funkcím, pro které má potřebná oprávnění. Ve smyslu vytvářeného systému je autorizace řešena na základě uživatelských rolí, kde každá role má předem specifikovaná oprávnění blíže popsané zde 4.1.2. Pro celkovou správu uživatelů a jejich rolí byl implementován administrátorský panel, skrz který je možné pozvat nové uživatele, upravovat jejich informace a především jim přidělit žádoucí role. Nad rámec současné implementace by mohlo být žádoucí implementovat více granulovanou zprávu pravomocí pro vyšší míru přizpůsobení přístupu pro jednotlivé zaměstnance. Na podobnou implementaci je systém připraven a bylo by možné ho tímto způsobem rozšířit. Pro jednotlivé uživatele by pak bylo možné vybrat konkrétní pravomoce a ne jen ty spojené s jednotlivými rolemi.

Autorizace jako taková je ve vytvářeném systému implementovaná hlavními **dvěma způsoby**. K prvnímu patří ověření uživatele před přístupem na každou stránku v systému skrz takzvané “middleware”, neboli prostřední vrstva tvořící mezistupeň mezi klientem a serverem. Konkrétní middleware je možné specifikovat na úrovni jednotlivých stránek a pro různé stránky je možné specifikovat jiné middleware. Pokud uživatel přistoupí na takovou stránku neboli provede *GET request*, před provedením požadavku proběhne ještě vykonání specifikované middleware funkce, ve které je možné provést libovolnou akci, jako např. číst či upravit http hlavičky nebo právě provést autentizaci a autorizaci uživatele.[20] Na většinu stránek k tomu je využito již zmíněné funkce *useSupabaseUser*, která ověří přihlášení uživatele a platnost jeho autentizačního tokenu. Pokud je třeba navíc ověřit uživatelskou roli, je využito již popsané funkce *getUser* na straně serveru dostupné skrze API *context* pro získání dat uživatele z databáze.

Druhým způsobem je ověření uživatele na straně serveru při provádění většiny požadavků a při volání API. Nejprve jsou získána data uživatele a pokud je pro vyžadovanou akci potřeba vyšší role, je provedena kontrola, zda ji objekt uživatele obsahuje. Pokud není možné získat uživatele na základě požadavku (např. HTTP hlavička neobsahuje platný jwt token), je vrácena odpověď s chybovým kódem *401 UNAUTHORISED*. Pokud uživatel je získán, ale nemá potřebnou roli, je vrácena odpověď s chybovým kódem *403 FORBIDDEN*.

5.3 Získávání a agregace dat z veřejných databází

Mezi významnou část implementace systému patřila agregace dat z externích zdrojů a jejich efektivní zobrazení a vhodné využití pro zjednodušení vybraných procesů advokátní kanceláře. Implementace této části se řídila návrhem popsaným v sekci 4.2. V této části je kladen větší důraz na popsání konkrétních způsobů získávání a zpracování dat a jejich ukládání. Dále je zde popsáno, jakým způsobem jsou získané informace prezentovány uživateli, jak s nimi může v rámci systému pracovat a jaké (pracovní) činnosti mu to může ulehčit jakožto člověku pohybujícím se v právní praxi.

5.3.1 Modul pro agregaci dat

Jelikož systém komunikuje současně s několika externími systémy, ze kterých získává odlišná data v různých formátech, bylo žádoucí vytvořit zastřešující modul, který bude získaná data z jednotlivých systémů dále zpracovávat do jednotné formy. Pro tento účel tak byl vytvořen modul *dataAggregator*. Tento modul implementuje funkce pro zpracování dat získaných po-

mocí dílčích modulů pro komunikaci s jednotlivými vybranými externími zdroji, které jsou také popsány v následujících podkapitolách. Tyto dílčí moduly vrací data ve 2 formátech. Buďto ve formátu celých HTML stránek, které je potřeba parsovat pro získání požadovaných informací, jelikož ne všechny z veřejných databází nabízí webové služby pro jednoduché získání dat ve strukturované formě nebo v strukturovaném XML formátu, ze kterého jsou též získávány relevantní informace v lépe čitelné formě. Pro parsování dat je využito dvou externích knihoven a to *node-html-parser*⁸ pro parsování HTML obsahu a *xml2json*⁹ pro parsování XML a převod na požadovaný JSON tvar. Veškeré získávání externích dat je tak prováděno skrz tento modul.

Modul *dataAgregator* vrací veškerá data již v jednotném JSON¹⁰ formátu, který je čitelný a jednoduše se s ním dále pracuje skrze celou aplikaci vzhledem k využití jazyka JavaScript, na syntaxi jehož objektů je formát JSON založen. Tento formát je v dnešní době velmi rozšířený pro přenos dat mezi systémy a pro tvorbu REST APIs avšak žádná ze stávajících veřejných databází nenabízí přímý přístup k datům v tomto formátu.

Zveřejnění API přístupových bodů pro získávání dat v tomto formátu dalšími externími systémy by tak mohlo být zároveň přínosné pro další vývojáře, jelikož by si mohli ušetřit čas s parsováním a zpracováním dat z původních zdrojů. Byly tak vytvořeny tři API koncové body, které jsou přístupné zvenčí systému a nabízejí jednoduché rozhraní pro získání vybraných dat z koncových databází v předzpracovaném a jednotném JSON formátu.

5.3.2 Informace o ekonomických subjektech

Pro získávání vybraných dat z obchodního rejstříku a živnostenského rejstříku bylo využito popsaného systému ARES 2.1.5. Webové služby tohoto systému byly využity pro získání všeobecných informací o ekonomických subjektech a pro konkrétní výpisy z obchodního rejstříku a rejstříku živnostenského podnikání. Výhodou je získání všech těchto dat v jednotném XML formátu z jednoho zdroje a tak umožnění efektivnějšího zpracování. Získání všeobecných informací o subjektu je možné primárně na základě názvu firmy, jména podnikatele, nebo IČO spolu s dalšími možnými kritérii. Jedná se o dotaz nazvaný “Standard” a jeho bližší dokumentace je k dispozici v systému Ares¹¹.

Výsledkem dotazu jsou v případě existence záznamu základní informace o subjektu, jako obchodní název, IČO, datum vzniku, právní forma a adresa. Dále odpověď obsahuje příznaky existence záznamu v konkrétních rejstřících, což je důležitá informace pro získání dalších detailních informací o subjektu již z konkrétních rejstříků. Získání těchto základních dat je tak využito v aplikaci pro poskytnutí informace o možné existenci jakéhokoli záznamu o subjektu v některém z veřejných rejstříků a nabídnout přístup k získání detailních informací o vybraném subjektu již z konkrétních rejstříků, kde je evidován. Pokud je například vyhledáno jen podle jména, může být nalezeno velké množství subjektů. Uživatel je tak nabídnut seznam výsledků se základními informacemi, na základě kterých může vybrat konkrétní subjekt. O vybraném subjektu následně může získat detailní informace z několika hlavních rejstříků zároveň a přehledně na jedné stránce.

Podobná funkcionalita je uživateli nabídnuta při přidávání nového klienta či subjektu do systému, v případě že se jedná o právnickou či fyzickou podnikající osobu. Je vyhledán

⁸<https://www.npmjs.com/package/node-html-parser>

⁹<https://www.npmjs.com/package/xml2json>

¹⁰<https://www.json.org/>

¹¹https://www.info.mfcr.cz/ares/ares_xml_standard.html.cz

její záznam v obchodním či živnostenském rejstříku a pokud uživatel volbou potvrdí, jsou informace o subjektu jako celý název, IČO a adresa předvyplněny do formuláře.

5.3.3 InfoSoud a InfoJednání

Z těchto stránek jsou získávány informace o soudních řízeních a nařízených soudních jednáních jak bylo blíže popsáno v 2.2. Funkce pro komunikaci s těmito systémy jsou implementovány v modulu *ejusticeCommunicator*. Postup získávání dat je velmi podobný pro obě stránky a je zde tak detailněji popsán konkrétní postup pro infoSoud. Postup pro InfoJednání probíhá obdobně, jelikož základní parametry dotazu jsou stejné (až na místnost jednání) a struktura výsledné stránky je také podobná té infoSoudu. Tyto stránky nenabízejí žádný přístup k datům pomocí využití webových služeb. Je tak nutné získat celou *HTML* stránku pomocí běžného *GET* požadavku se stejnými parametry, jako kdybychom ručně vyplnili formulář skrz originální uživatelské webové rozhraní a následně z ní **extrahovat** požadované informace.

Pro to je potřeba sestavit URL adresu dotazu s následujícími parametry:

- Kód krajského soudu, kde je případ řešen,
- Kód okresního soudu,
- Spisová značka (číslo senátu, druh věci, běžné číslo, ročník) případu,
- Jednací síň (jen u infoJednání).

Kódy soudů byly získány z původní stránky a uloženy jako konstanty v souboru *composables/constants.js*. Tyto údaje jsou získány od uživatele při přidání spisové značky ke spisu. V dotazu je též vyžadován parametr *spamQuestion*, který je však neměnný a v době vývoje systému byl stanoven na konstantní hodnotu 23. Pokud by byla tato hodnota poskytovatelem služby v budoucna změněna, bylo by potřeba upravit i tento dotaz, či změnit přístup k získávání dat. Výsledná URL adresa může potom vypadat například takto:

```
https://infosoud.justice.cz/InfoSoud/public/search.do?type=spzn&krajOrg=${
  ↪ regionCourt}&org=${districtCourt}&cisloSenatu=${senateNumber}&
  ↪ druhVec=${topicType}&bcVec=${topicNumber}&rocnik=${year}&
  ↪ spamQuestion=23
```

Obrázek 5.4: Příklad URL adresy pro získání dat z infoSoud.

Vrácený výsledek ze služby je opět zpracován v modulu *dataAgregator*, kde je provedeno parsování *HTML* obsahu pomocí funkcí z již zmíněné knihovny *node-html-parser*.

V prvním kroce parsování je využito knihovní funkce *parse*, která získaný *HTML* dokument rozparsuje a převede do strukturované podoby, se kterou je následně možné pracovat podobně jako se skutečným *DOM* stromem [57] a efektivně přistupovat k jednotlivým elementům pomocí selektorů podobným *CSS* selektorům, jak je znázorněno v následující ukázce.

```
const root = parse(page);
const bodyBannerTd = root.querySelector("td.body-banner");
```

Výpis 5.1: Ukázka kódu pro získání elementu *td* se třídou *body-banner*

Nejprve je proveden pokus o získání elementu obsahující případný chybový status dotazu, a pokud obsahuje text, je tento text vrácen jako chybová odpověď a zobrazen uživateli. Může se jednat například o chybu jako "neexistující spisová značka" či špatný formát některého z parametrů.

Pokud je výsledek v pořádku, je zobrazen stav řízení a tabulka průběhu řízení s jednotlivými událostmi. Pro zobrazení detailu události je nutné prokliknout se na stránku detailu, kde jsou kromě názvu události a data další bližší informace, jako případné místo a čas jednání, způsob rozhodnutí a další.

Cílem tedy bylo získat z této stránky veškeré události řízení, včetně jejich detailů. Jelikož jsou data denně aktualizována a požadavkem na systém je, aby sledoval případné změny v řízení, které jsou znázorněny jednotlivými událostmi, je potřeba informace o řízení a jeho událostech uložit do databáze. Jedině tak může být při opětovném dotázání porovnáním zjištěno, zda došlo ke změně.

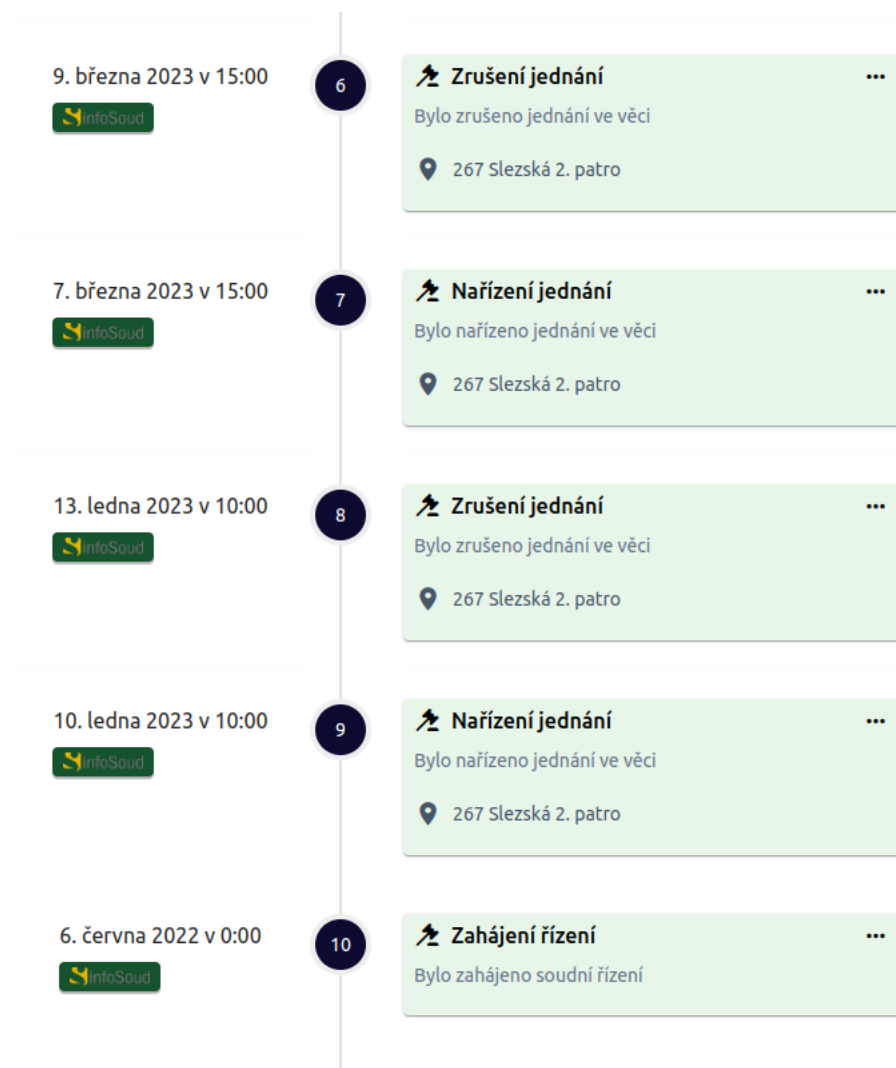
Při prvním přidání spisové značky jsou tak automaticky rovnou získány a vyparsovány veškeré informace o řízení a uložena do databáze do tabulky řízení a stejně tak veškeré události tohoto řízení s jejich detailními informacemi. Ta jsou uložena v tabulce událost řízení, kde každý záznam odkazuje na původní řízení.

Při opětovném získání informací pro již uložené řízení je po zpracování dat provedeno porovnání s daty z databáze na základě událostí. Pokud je v nově získaných datech řízení nová událost/nové události, jsou tyto také uloženy do databáze a navíc je vytvořeno upozornění pro uživatele, s informací že došlo ke změně tohoto soudního řízení. Ukázka notifikací je zobrazena na obrázku 5.6. Pro každou událost řízení je též vytvořena aktivita, které se zobrazují v časové ose, která je k vidění na obrázku 5.5. Funkce pro tuto aktualizaci dat jsou implementovány v modulu **dataUpdater**. Ten využívá plánovací knihovny *node-schedule*¹² pro provádění pravidelné kontroly a aktualizace dat jednou denně ve vybraný čas. Tato funkcionalita je implementovaná v modulu *scheduler*.

Testování funkcionality bylo provedeno na skutečných probíhajících soudních řízeních s přiřazenými spisovými značkami, které je možné nalézt díky stránce nesoucí název "Právnícké výpočty"¹³.

¹²<https://www.npmjs.com/package/node-schedule>

¹³<https://www.pravnicke-vypocty.com/>



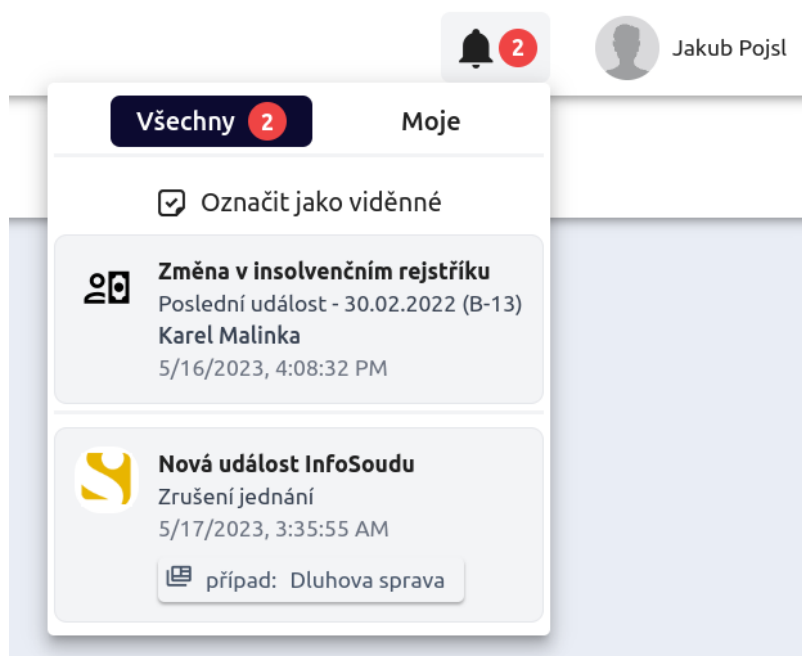
Obrázek 5.5: Snímek obrazovky s pohledem na část časové osy v detailu spisu zobrazující nastalé události v rámci řízení

5.3.4 Insolvenční rejstřík

Insolvenční rejstřík byl zvolen jako poslední v řadě veřejných databází pro agregaci dat, vzhledem k jeho důležitosti v právní praxi. Změny v insolvenčních řízeních mohou mít vliv na klienty advokátních kanceláří a případy s nimi řešenými a data z něj jsou tak často využívána, jak bylo blíže popsáno v části 2.1.2. Požadavkem na systém tak bylo, aby umožňoval automatické získávání dat o insolvenčních řízeních a upozorňoval na případné nastalé změny. Nabízela se tedy možnost využití nabízeného webového rozhraní vytvořeného pro tento účel, popsané ve druhé části podkapitoly 2.1.2. Po bližší analýze se však rozhraní nezdálo jako příliš vhodné pro vyžadované případy užití. Rozhraní umožňuje získat buďto pouze základní informace, nebo výčty všech událostí pro všechna řízení pro vlastní rekonstrukci databáze. Jelikož mezi specifikované požadavky patří spíše vyhledání řízení podle subjektu, získání jeho detailního průběhu a sledování změn, ani jedna z možností se nejevila jako ideální pro efektivní splnění těchto požadavků. Rozhodl jsem se tedy opět pro implementaci získávání dat vlastním způsobem.

Postup pro získání dat je tak obdobný jako v předchozím případě. Pomocí odesílání *GET* požadavků s odpovídajícími parametry je vždy získána celá *HTML* stránka, jako při využití přístupu skrz rozhraní ve webovém prohlížeči. Získaná stránka je zpracována a je provedena extrakce požadovaných informací. Pro získání požadovaných dat byly implementovány dvě hlavní funkce. První z nich je pro vyhledání subjektů se záznamem v rejstříku na základě jména a příjmení, identifikačního čísla, či názvu firmy. Druhá z nich je pro získání detailních informací o konkrétním řízení včetně všech událostí. Funkce pro komunikaci s insolvenčním rejstříkem jsou implementovány v modulu *isirCommunicator*. Získaná data jsou poté dále opět zpracována v modulu *dataAggregator*. Průběh odpovídá znázornění na sekvenčním diagramu v části návrhu 4.2.2. Tyto funkce byly také zpřístupněny skrze externí API koncové body *searchDebtorsISIR* a *proceedingDetailISIR*. Je však nutno poznamenat omezení insolvenčního rejstříku, které limituje maximální počet dotazů denně na tři tisíce a maximálně 50 dotazů za minutu. Tento přístup by tedy nebyl využitelný pro velkokapacitní zpracování dat, avšak je vhodný pro účely systému, kde by tento limit neměl být přesažen. Pro vyhnutí se překročení limitům bylo implementováno vlastní omezení využití a při případném rozsáhlejší škálování systému by mohlo být využito přístupu s vytvářením vlastní rekonstrukce celé databáze.

Uživatel má možnost vyhledání subjektů a při zvolení subjektu je mu zobrazen detailní přehled řízení včetně všech událostí. Dále má uživatel možnost nastavit sledování tohoto řízení. Tím je vytvořen nový záznam subjektu v systému a nový záznam řízení insolvenčního rejstříku, které je se subjektem propojeno. Díky uložení do databáze je tak následně možné pravidelně sledovat nastalé změny jako v případě soudních řízení. K tomu je opět využito modulu *scheduler* pro pravidelné provádění akcí a modulu *dataUpdator*, který provádí porovnání záznamů v databázi s nově získanými daty a v případě změn aktualizuje data a vytvoří upozornění pro uživatele. Přehled řízení a zobrazené upozornění jsou k vidění na následujících obrázcích.



Obrázek 5.6: Snímek obrazovky s pohledem na detail sekce upozornění s novými upozorněními na změnu v insolvenčním rejstříku a na novou událost z infoSoudu

The screenshot shows a web interface for searching economic subjects. On the left, there is a search form with fields for IČO, Příjmení/Název (filled with 'Malinka'), and Jméno. Below the form, it shows search results for 'Karel Malinka' with details like address and status. On the right, there is a section for 'Toto řízení je sledováno a propojeno s následujícími subjekty' (This proceeding is monitored and linked to the following subjects), with a search bar and a profile icon for 'Karel Malinka'. Below this, there is a table of events (událostmi) for the insolvency proceeding, categorized by 'Oddíl' (Section).

Vyhledat ekonomický subjekt
Podle IČO, jména či názvu firmy (bez diakritiky)

IČO

Příjmení/Název
Malinka

Jméno

Zrušit Vyhledat

Počet výsledků:
OR a RŽP: 42 ISIR: 5

Karel Malinka
Brno, Míru 22, PSČ 760 01
Odškrtnutá - skončená věc

Antonín Malinka
Brno, náměstí Svobody 17, 602 00
Odškrtnutá - skončená věc

František Malinka, IČO: 37943417
Ostrava, Masarykovo náměstí 1, 702 00
Povoleno oddlužení

Petr Malinka
Zlín, náměstí Míru 1, 760 01
Povoleno oddlužení

David Malinka
Pardubice, náměstí Republiky 1, 530 02
Povoleno oddlužení

Insolvenční rejstřík

Toto řízení je sledováno a propojeno s následujícími subjekty

Dlužník

Karel Malinka

Karel Malinka
Aktuální stav: Odškrtnutá - skončená věc
Spisová značka: KSBR 12 INS 11215 / 2017 Krajského soudu v Brně
Datum poslední zveřejněné události: 30.03.2023 (B-13)

Oddíl A		Oddíl B		Oddíl C		Oddíl D		Oddíl P	
Index	Okamžik zveřejnění	Čas	Popis	Dokument	Vedlejší dokument	Datum právní moci	Senátní značka VS/NS		
1.	26.07.2016	09:46	Insolvenční návrh spojený s návrhem na povolení oddlužení	plný text (1157 kB)	-				
2.	26.07.2016	09:46	Pověření asistenta soudce/VSÚ	plný text (259 kB)	-				
3.	26.07.2016	09:46	Vyhláška o zahájení insolvenčního řízení	plný text (280 kB)	plný text (347 kB)				
4.	26.07.2016	10:27	Návrh - přílohy	plný text (3875 kB)	-				
5.	09.09.2016	09:32	Žádost o určení osoby správce	plný text (230 kB)	-				
6.	13.09.2016	09:20	Opatření o určení osoby správce	plný text (150 kB)	-				
7.	16.09.2016	10:13	Usnesení o úpadku spojené s povolením oddlužení	plný text (340 kB)	plný text (372 kB)	19.09.2016			
8.	03.10.2016	13:01	Odvolaň jiné	plný text (372 kB)	-				
9.	10.10.2016	08:50	Výzva	plný text (276 kB)	plný text (200 kB)				
10.	10.10.2016	15:53	Opravné usnesení	plný text (292 kB)	plný text (363 kB)	02.11.2016			
11.	20.10.2016	09:46	Sdělení insolvenčního správce	plný text (182 kB)	-				

Obrázek 5.7: Snímek obrazovky s pohledem na výsledky vyhledávání subjektu a zobrazení detailu insolvenčního řízení s událostmi

5.4 Nahrávání a správa dokumentů

Tato kapitola se zaměřuje na popis implementace funkcionality pro nahrávání a správu dokumentů v rámci vyvíjeného systému. Jak bylo z dosavadní analýzy zřejmé, práce s dokumenty je v oblasti práva velmi častá. Bylo tak třeba realizaci dobře rozmyslet, aby byla funkcionality efektivní a splňovala potřeby a očekávání uživatelů. Je zde tak přiblíženo, jaké možnosti reprezentace souborové struktury byly zvažovány a jaký přístup byl nakonec zvolen a proč. Diskutované jsou také možnosti napojení osobního externího cloudového úložiště, jako je google drive či MS one drive a využití cloudového úložiště Supabase storage¹⁴ pro ukládání libovolných souborů za použití API. Nakonec jsou popsány konkrétní implementované funkce pro práci s dokumenty a složkami a způsob ukládání dat.

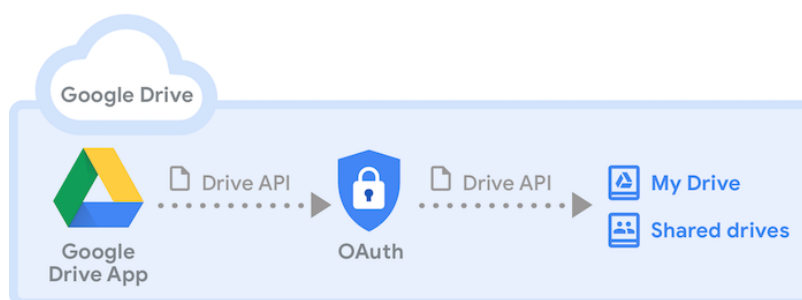
Výsledná funkcionality pro správu dokumentů je na základě specifikovaných a analyzovaných požadavků koncipována jako zjednodušený souborový systém, který je intuitivně provázaný s jednotlivými klienty a spisy advokátní kanceláře. Z analýzy procesů advokátní kanceláře bylo zjištěno, že dokumenty jsou v elektronické formě převážně organizovány tímto způsobem pomocí využití nativního souborového systému počítače. Vytvoření ob-

¹⁴<https://supabase.com/docs/guides/storage>

dobné možnosti v implementovaném systému tak umožní uživatelům s dokumenty pracovat podobným způsobem, jako jsou zvyklí, avšak s výhodou přímého provázání s konkrétními klienty a spisy a s navázáním na další funkce systému, jako je například verzování, což umožní efektivnější práci s dokumenty. Díky uložení dat v cloudovém úložišti a multiplatformní dostupnosti vyvíjeného systému budou navíc soubory přístupné z libovolného zařízení s přístupem k internetu.

5.4.1 Využití externího úložiště dat

Jako první možností pro umožnění správy souborů v systému se nabízelo zabudování již některého z existujících řešení. Zvažoval jsem konkrétně možnosti využití služby Google Drive¹⁵ a konkurenční služby One Drive¹⁶ od společnosti Microsoft. Obě služby nabízejí velmi propracovaná řešení pro online úložiště souborů včetně dostupného a dobře zdokumentovaného API pro využití vývojáři třetích stran. Po průzkumu obou řešení a pročtení částí dokumentace jsem ale usoudil, že by bylo pro obě platformy potřeba věnovat poměrně značné úsilí na jejich zakomponování do systému, které by mohlo obnášet komplikace. Bylo by potřeba vyřešit nastavení vývojářského účtu, zajistit správu a implementaci dodatečné autentizace a implementaci pro získání autorizace uživatelů pro přístup k jejich již existujícímu úložišti a vybraným adresářům na těchto platformách. Pro zajištění požadované funkcionality by mimo jiné ve výsledku bylo nutné pro velkou část stejně vytvořit vlastní implementaci, aby bylo možné zajistit požadované provázání s dalšími objekty v systému jako jsou klienti a spisy a umožnit intuitivní práci s dokumenty přímo v systému bez odkazování uživatele na externí portál daného úložiště. Problematické by mohlo být také udržování konzistence v přístupu na základě pravomocí, které se v implementovaném systému řídí uživatelskými rolemi, avšak v externím úložišti můžou být libovolně upraveny nezávisle a jiným způsobem. [49, 58] V neposlední řadě uživatelé nemusí účty na těchto platformách vlastnit a nemusí mít zájem o jejich vytváření pro zajištění funkcionality správy dokumentů, kterou už od samotného systému mohou očekávat přímo.



Obrázek 5.8: Diagram Google Drive API. Převzato z: [<https://developers.google.com/drive/api/guides/about-sdk>]

Celkově by se ale jednalo o jistě zajímavou možnost budoucího rozšíření, kterou by bylo možné uživatelům nabídnout navíc, vedle zabudované základní funkcionality pro správu

¹⁵<https://www.google.com/drive/>

¹⁶<https://www.microsoft.com/en-us/microsoft-365/onedrive/online-cloud-storage>

dokumentů. Uživatelům, kteří takového úložiště již využívají by to umožnilo jednoduší import jejich souborů do systému a možnost alternativního přístupu k dokumentům z rozhraní daného úložiště. Uživatelé by měli nezávislou kontrolu nad tímto svým úložištěm včetně jeho velikosti. Tyto rozhraní samy o sobě také nabízejí pokročilejší funkcionalitu pro práci se soubory a nabízí i možnost jejich přímého upravování online ve svých propracovaných editorech¹⁷, které navíc umožňují spolupráci více lidí v reálném čase. Uživatelům by to tak bezesporu přineslo užitečné výhody.

5.4.2 Možnosti implementace souborového systému

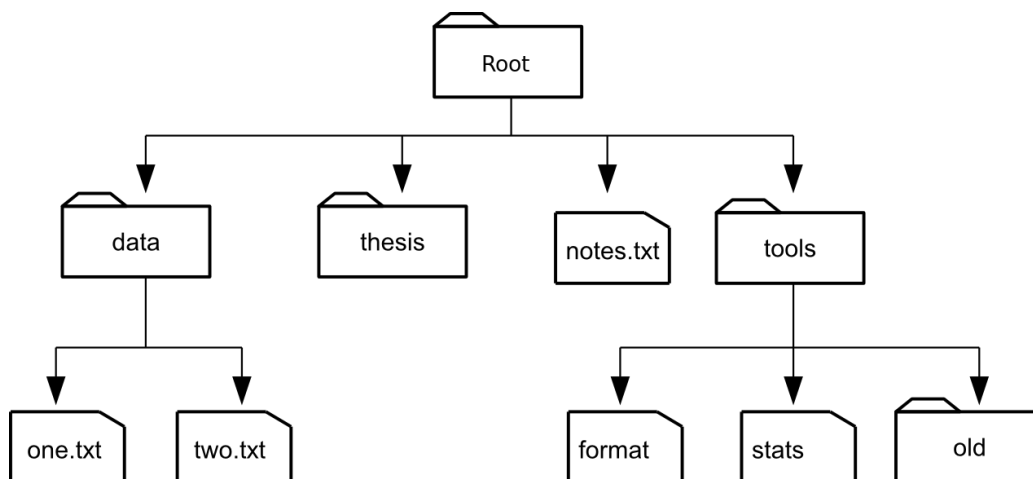
Po zvážení dostupných možností a požadavků jsem se rozhodl pro implementaci vlastního rozhraní pro umožnění správy dokumentů v podobě zjednodušeného souborového systému navázaného na jednotlivé klienty i spisy. Využito bylo vlastní *postgreSQL* databáze v kombinaci s **cloudovým úložištěm** souborů Supabase Storage¹⁸ od poskytovatele Supabase a vytvoření vlastních komponent uživatelského rozhraní pomocí zvolených frontend technologií.

Typickou vlastností souborového systému je, že uživateli umožňuje vytvářet **hierarchickou strukturu** libovolně zanořených složek a souborů a tak efektivně organizovat dokumenty podle vlastního uvážení. Pro replikování této vlastnosti bylo potřeba zvážit jakým způsobem bude této funkcionality možné dosáhnout s pomocí využívaných technologií.

Jelikož hierarchie složek a dokumentů je určitá stromová struktura, jedním ze způsobů implementace by mohla být forma dvoustraně vázaného seznamu. Kořenový adresář představuje kořen stromu, vnořené adresáře je možné reprezentovat jako jeho přímé potomky, kde každý z nich má k dispozici odkaz na svůj rodičovský adresář a dokumenty vystupují jako listy stromu. Tento způsob reprezentace umožňuje snadnou manipulaci se soubory a složkami jelikož v takové struktuře je jednoduché přidávat, mazat či přesouvat uzly se zachováním integrity dat. Z pohledu *postgreSQL* databáze může být taková struktura reprezentovaná 0 ku N vazbou tabulky složka na sebe samou s vazbou 0 ku N do tabulky dokumentů. Neboli složka může obsahovat libovolné množství jiných složek a dokumentů. Složka i dokument pak obsahuje cizí klíč rodič, odkazující na složku ve které se nachází, který může být pro složku prázdný (v takovém případě se jedná o kořenový adresář). Graficky znázorněno v ER diagramu v sekci 4.3.2.

¹⁷<https://www.google.com/docs/about/>

¹⁸<https://supabase.com/docs/guides/storage>



Obrázek 5.9: Příklad chématu adresářové struktury. Převztao z: [\[http://swcarpentry.github.io/2014-04-14-wise/novice/shell/01-filesystem.html\]](http://swcarpentry.github.io/2014-04-14-wise/novice/shell/01-filesystem.html)

Alternativní metodou implementace stromové struktury je pomocí ukládání unikátního řetězce obsahujícího cestu od kořene stromu až k danému uzlu, kde uzlem může být soubor nebo adresář. Například *Root/složka1/soubor1*. Tento přístup by mohl umožnit jednodušší tvorbu SQL dotazů, ale operace jako přesun či přejmenování by byly komplikovanější, jelikož by bylo pokaždé třeba aktualizovat cesty všech závislých uzlů. Z pohledu databáze by byla implementace proveditelná pomocí tabulek složka a dokument, kde každý obsahuje jen svou unikátní cestu ve stromu jako string a není potřeba žádné vzájemné vazby.

Vzhledem k zvážení uživatelských potřeb a snadnosti implementace jsem se rozhodl pro první popsaný přístup, který se zdá vhodnější z důvodu jednoduššího provádění typických operací se složkami a dokumenty jako je jejich přesouvání a přejmenování. Implementace v databázi tak odpovídá prvnímu popsanému přístupu v této kapitole, který je též graficky znázorněn v ER diagramu. Dokumenty jsou zároveň nahrávány do cloudového úložiště Supabase Storage s využitím nabízeného API, které umožňuje jednoduše provádět základní operace jako je nahrávání, stahování, či mazání souborů libovolného formátu na základě unikátní url adresy souboru a existujícího autentizačního tokenu. Tento proces je blíže popsán v následující podkapitole spolu s frontend implementací a implementovanými funkcemi. Supabase storage bylo vybráno, jelikož je nabízený poskytovatelem Supabase, jehož služeb je využíváno i pro autentizaci a hosting databáze. K těmto službám nabízí zdarma využití úložiště souborů do velikosti 50GB s možností placeného škálování. Nebylo tak potřeba implementovat odlišnou autentizaci pro úložiště. Implementované funkce by bylo možné ale jednoduše napojit i na API jiného poskytovatele cloudového úložiště, pokud by v budoucnu byla vyžadována změna.

5.4.3 Implementace funkcí pro práci se soubory a adresáři

V této části se podrobněji zaměříme na implementaci jednotlivých funkcí pro správu dokumentů, které systém nabízí včetně popisu vybraných částí implementace a uvedení příkladů. Nejprve je stručně popsán celkový princip implementace a zpřístupnění souborového systému uživateli a jeho provázání s klienty a spisy. Znázorněna je tvorba frontend komponent za využití *vue.js* a jejich využití pro tvorbu **UI souborového systému**. Uvedeny jsou také ukázky využití Supabase Storage API. Veškeré pomocné funkce související se správou dokumentů jsou implementovány v modulu pro správu stavu *documentStore*.

Dokumenty jsou v systému přístupné vždy z karty klienta či karty spisu pomocí horní subnavigace. Při každém vytvoření nového klienta či spisu je automaticky vytvořena i jeho kořenová složka pro dokumenty, která je s ním propojena a nemůže být odstraněna. V případě složky spisu je její nadřazenou složkou složka klienta. Nadřazenou složkou složky klienta je defaultně kořenová složka celé advokátní kanceláře, která je vytvořena spolu s instancí nové advokátní kanceláře.

Při otevření záložky dokumenty je na základě spisu či klienta zjištěna i přiřazená složka a je získán a zobrazen její případný obsah, se kterým je možné dále interagovat. Při otevření každé složky je provedeno volání API *folders/get-content* ve kterém je implementováno dotázání databáze na získání obsahu složky spolu s rekurzivním získáním cesty od složky ke kořenu. Získání obsahu složky zahrnuje získání všech přímo vložených složek a dokumentů, včetně všech případných verzí dokumentů.

Vytváření a přejmenování složek

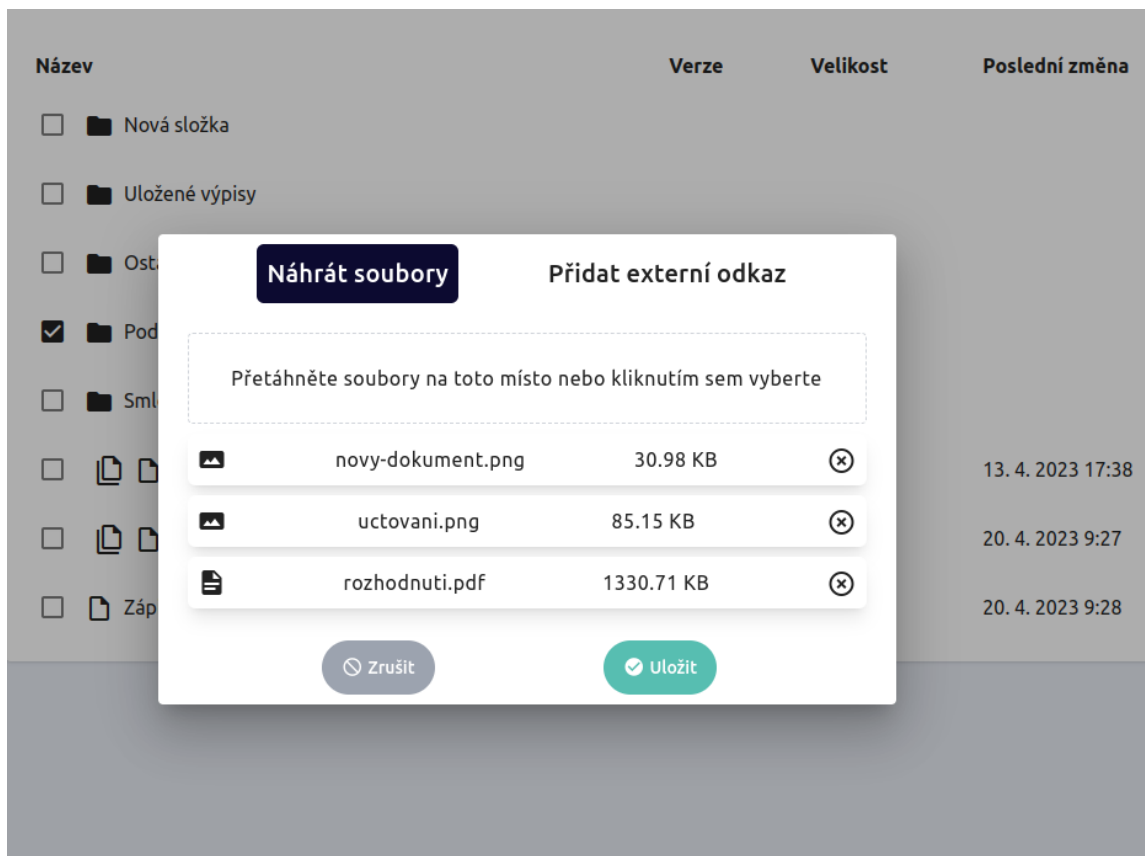
Jedná se o nejzákladnější funkce pro umožnění organizace dokumentů podle vlastních preferencí a je na ni zvyklý snad každý uživatel počítače. Při vytváření složky je možné zadat jméno, které nesmí kolidovat se jménem jiné složky se stejným umístěním a stejně tak při přejmenování složky, pro což je implementována kontrola.

Nahrávání dokumentů a jejich verzí

Jedná se o jednu z hlavních požadovaných funkcionalit ohledně správy dokumentů. Uživatel může do systému a vybrané složky nahrát **libovolný soubor** či několik souborů o maximální velikosti jednoho souboru 50MB jelikož to je limit bezplatné verze nabízené Supabase Storage. Funkcionalita je dostupná skrz vyskakovací okno k vidění na obrázku 5.10. Při přidání nového souboru je soubor nahrán do tohoto úložiště pomocí funkce *upload* klientské knihovny se znázorněním průběhu **nahrávání**, následně je vytvořen záznam dokument ve vlastní databázi s uložením url adresy odkazující do úložiště, která je využita i pro **stažení souboru**. Při přidávání souboru je provedena kontrola existence souboru se stejným názvem v dané složce a v případě shody je zobrazeno varování a nabídnuta možnost vytvoření nové verze existujícího dokumentu.

Pro zachování alespoň částečné možnosti propojení s **externími úložišti** nebo přiložení souboru libovolné velikosti, byla implementována možnost vytvoření souboru s přidáním **vlastního odkazu** na místo uložení souboru. Toto tak umožní uživateli mít případné soubory z více zdrojů odkázané z jednoho centrálního místa přímo pod spisem či složkou klienta.

Ke každému dokumentu je možné přidat libovolné množství dalších **verzí**. Proces přidání nové verze je podobný jako přidání samostatného dokumentu. V databáze pouze tento dokument odkazuje na původní dokument a má automaticky přiřazeno číslo verze.



Obrázek 5.10: Snímek obrazovky modálního okna s formulářem pro výběr a nahrání souborů ze zařízení

Přesouvání dokumentů a složek

Toto je důležitá a očekávaná funkcionality pro umožnění jednoduché **organizace souborů** a složek a byla tak implementována i v rámci vytvářeného systému. Vzhledem k volbě reprezentace pomocí **vázaného seznamu**, logiku operací bylo možné implementovat poměrně jednoduše. Při přesunu složky či souboru, stačí aktualizovat vazbu na nadřazenou složku na hodnotu složky do které bylo provedeno přesunutí. Nutné bylo přidání **kontroly** pro zamezení vložení složky do sebe samotné nebo do jakékoli jiné v ní zanořené složky, jelikož jinak by došlo k vzniku cyklu a oddělení této větve od zbývajících částí stromu adresářů. Jednou z možností přesunutí souborů a složek je jejich **vyjmutí** a opětovné **vložení** v místě požadovaného umístění.

Z běžných souborových systémů i online úložiště jsme kromě vyjmutí a vkládání či kopírování zvyklí využívat pro manipulaci se soubory možnost jejich **přetažení** do požadované cílové lokality. Pro zachování konzistentní uživatelské zkušenosti jsem se rozhodl tuto funkcionality implementovat také. Využito bylo nativního rozhraní webového prohlížeče *Drag & Drop API*¹⁹, jehož události mohou být zaznamenány a na jejich základě může být vyvoláno provedení libovolné funkce.

Mezi tyto události patří například:

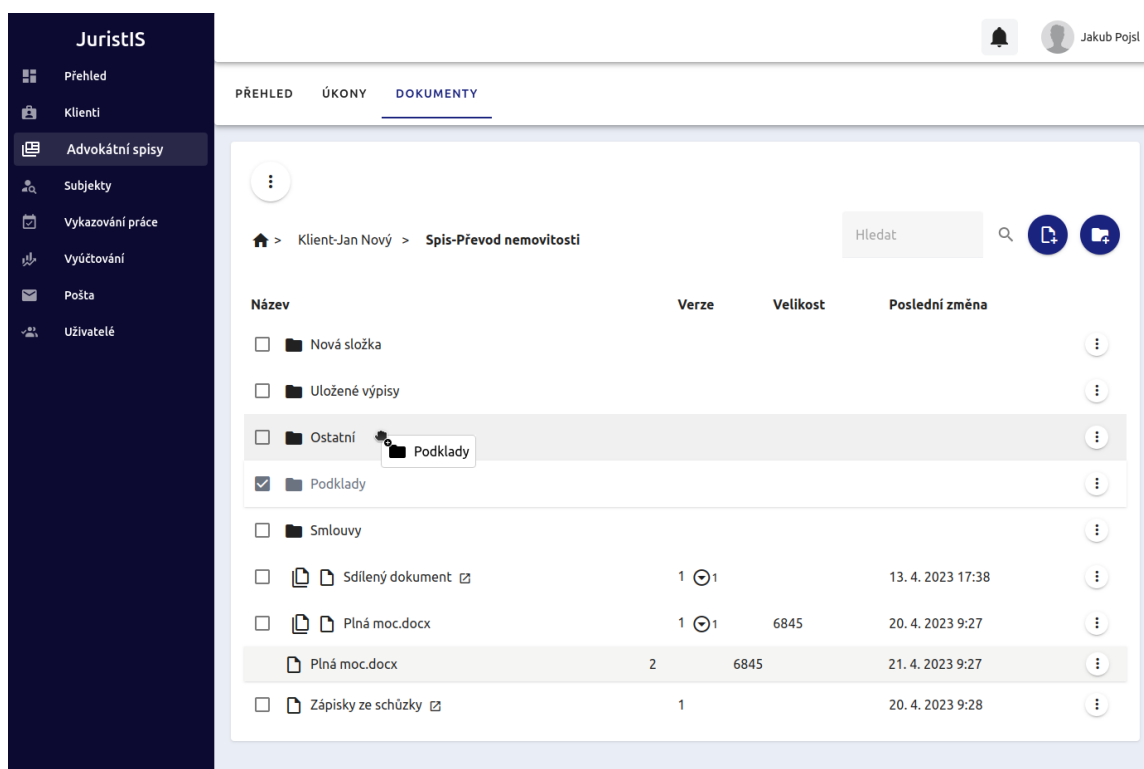
- *dragstart* - nastává, když uživatel začne přetahovat objekt,

¹⁹https://developer.mozilla.org/en-US/docs/Web/API/HTML_Drag_and_Drop_API

- *dragenter* - nastává, když přetahovaný objekt vstoupí do cílové oblasti,
- *drop* - nastává, když uživatel pustí přetahovaný objekt na cílové oblasti,
- *dragend* - nastává, když drag and drop proces skončí.

Při vzniku těchto událostí jsou provedeny odpovídající funkce například pro změnu vzhledu přesouvaného souboru, zvýraznění složky pro vložení či funkce pro přesun po vložení do cílové složky. Náhled uživatelského rozhraní s demonstracím této funkcionality je k vidění na následujícím obrázku 5.11.

Samozřejmě je funkcionality provádění operací s **několika složkami** a dokumenty zároveň, je tak implementována možnost **označení** libovolného množství složek a dokumentů a jejich **současné přesunutí** či vyjmutí a následné vložení. Vybrané soubory jsou uloženy do reaktivního pole a pro vyjmuté soubory je vytvořena stavová proměnná *clipboard* v modulu *documentStore*.



Obrázek 5.11: Snímek obrazovky stránky dokumentů ve spisu klienta s demonstrací funkcionality drag & drop

5.5 Správa pošty a datové schránky

Po domluvě s vedoucím práce jsem se pro hlavní způsob implementace evidence pošty rozhodl pro napojení systému na systém **datových schránek**. Tento systém je v právní sféře velmi hojně využíván a to hlavně z důvodů legálního uznávání datových zpráv jakožto oficiální možnosti komunikace s úřady, soudy a dalšími veřejnoprávními institucemi. **Informační systém datových zpráv** (dále jako ISDS) navíc nabízí propracované webové *SOAP* rozhraní pro přístup ke službám datových schránek a možnost využití rozhraní aplikacemi

třetích stran jako je právě vyvíjený systém. Rozhraní disponuje kvalitní dokumentací a bylo podle ní možné implementovat základní funkcionalitu pro práci s datovými zprávami jako je přihlášení k datové schránce, získání seznamu přijatých zpráv, či zobrazení detailu konkrétní zprávy. Během implementace této funkcionality bylo třeba brát v potaz, že okamžik doručení datové zprávy je z právního pohledu vnímáno jako legitimní převzetí písemnosti a může tak znamenat započítání běhu různých lhůt a další konsekvence. Je potřeba také bezpečně nakládat s přihlašovacími údaji uživatele. Kromě datový zprávy je implementována i základní schránka pro zaznamenávání libovolné vlastní korespondence. V následujících kapitolách je popsán postup při integraci systému datových zpráv a jaké operace pro správu pošty byly implementovány. Funkce pro komunikaci s ISDS jsou implementovány v modulu *isdsCommunicator*.

5.5.1 Autentizace s správa schránek

Systém umožňuje uživatelům přidat libovolný počet svých **datových schránek** na základě **uživatelského jména a hesla**. Všechny přidání schránky uživatele jsou načteny v rozhraní pošty a uživatel mezi nimi může jednoduše přepínat. Pro umožnění přidání nové datovým schránky bylo třeba implementovat formulář pro získání přihlašovacích údajů od uživatele. Ta jsou po vyplnění odeslána pomocí zabezpečeného *HTTPS* požadavku na server skrz *API* koncový bod *messages/add-new-inbox*. Následně je v první řadě potřeba ověřit, že datová schránka s danými přihlašovacími údaji existuje a uživatel je k ní oprávněn přistoupit. K tomu byla vytvořena funkce *getOwnerInfoFromLogin*. Tato funkce vytváří *SOAP* požadavek obsahující autentizační hlavičku s uživatelským jménem a heslem, který je odeslán na server informačního systému datových zpráv.

Na základě jedné z doporučených metod přihlášení v dokumentaci ISDS je využito **Basic HTTP autentizaci**. Ta spočívá v tom, že uživatelské jméno a heslo jsou zakódovány do *Base64* a následně přidány do HTTP hlavičky jako autentizační token. Tento token je odeslán spolu se *SOAP* požadavkem na server ISDS.

Pokud jsou přihlašovací údaje platné, server vrátí informace o datové schránce ve formě XML dat, která jsou následně převedena do JSON formátu pro další zpracování a **uložení datové schránky** uživatele do databáze včetně jeho přihlašovacích údajů a hlavních informací o schránce jako je její id a název. Uložení těchto údajů umožní načtení všech schránek uživatele při přístupu do sekce pošty bez nutnosti opětovného přihlašování k jednotlivým datovým schránkám. [62]

5.5.2 Šifrování citlivých údajů

Jelikož od uživatele získává citlivá osobní data v podobě **přihlašovacích údajů** k jeho datové stránce, je třeba zajistit maximální zabezpečení těchto dat. Ukládání hesel v nezašifrované textové podobě, představuje vážné riziko pro bezpečnost uživatelů a je považováno za špatnou praxi, jelikož v případě úniku nebo kompromitace databáze by útočníci mohli snadno zneužít tyto citlivé informace. Byla tak implementována funkcionalita pro **zašifrování hesla** před jeho uložením do databáze pomocí vygenerovaného tajného klíče přístupného jen na serveru systému. Tento klíč je poté použit i pro dešifrování hesla před jeho použitím pro basic HTTP autentizaci uživatele při komunikaci s ISDS [60]. Procesem zašifrování je zajištěno vyššího zabezpečení hesla uživatele a sníženo riziko jeho potenciální kompromitace. Nyní je blíže popsán proces šifrování dat včetně použitého algoritmu

a využití modulu *crypto*²⁰ který je vestavěnou součástí *Node.js*. Jedná se o modul, který implementuje užitečné funkce pro podporu kryptografie.

Pro šifrování respektive dešifrování hesel byly implementovány funkce *encrypt()*, respektive *decrypt()*, které spolupracují s funkcí *getKey()* pro získání **šifrovacího klíče**.

Proces šifrování probíhá následovně:

1. Funkce *getKey()* vytváří šifrovací klíč pomocí algoritmu *PBKDF2* (Password-Based Key Derivation Function 2). [59] Tento algoritmus je široce používán pro generování klíčů z hesel a zahrnuje následující parametry:
 - a. Tajný klíč: Základní heslo, ze kterého je odvozen šifrovací klíč.
 - b. Sůl (salt): Náhodný řetězec, který zvyšuje odolnost proti slovníkovým útokům a odvození klíče pomocí tabulek.
 - c. Iterace: Počet opakování hašovacího procesu, který zvyšuje úroveň zabezpečení při generování klíče.
 - d. Délka klíče: Požadovaná délka šifrovacího klíče.
 - e. Hašovací algoritmus: Algoritmus, který je použit pro hašování, v tomto případě *SHA-256* (Secure Hash Algorithm 256-bit).
2. Po získání šifrovacího klíče je tento klíč použit v rámci funkce *encrypt()* pro zašifrování hesla. Pro šifrování je použit algoritmus *AES-256-GCM* (Advanced Encryption Standard s 256-bitovým klíčem a Galois/Counter Mode). Tento algoritmus je široce uznáván jako jeden z nejbezpečnějších symetrických šifrovacích algoritmů. *AES-256-GCM* kombinuje šifrování a autentizaci dat, což zajišťuje důvěrnost a integritu informací.
3. Proces šifrování začíná inicializací inicializačního vektoru (IV), který je náhodný řetězec o délce 16 bajtů. Inicializační vektor zajišťuje, že zašifrovaný text bude výrazně odlišný i při opakovaném šifrování stejných dat.
4. Šifrovací klíč a inicializační vektor jsou použity k vytvoření šifrovacího objektu pomocí funkce *crypto.createCipheriv()*.
5. Textová data (heslo) jsou zašifrována pomocí šifrovacího objektu a funkce.
6. Po zašifrování dat je získána autentizační značka (authTag) pomocí funkce *getAuthTag()* z třídy *cipher*. Autentizační značka je použita pro ověření integrity a autenticity zašifrovaných dat při dešifrování.
7. Zašifrovaná data, inicializační vektor a autentizační značka jsou spojeny do jednoho výstupního objektu typu *Buffer*. Tento objekt je pak uložen do databáze jako zašifrované heslo uživatele jako typ *Bytes*.

Proces dešifrování je v mnoha ohledech analogický procesu šifrování. Z databáze je získán zašifrovaný záznam hesla, který obsahuje inicializační vektor, autentizační značku a zašifrovaná data. Stejným způsobem, jakým byl získán šifrovací klíč, je získán klíč pro dešifrování. S využitím funkcí *crypto.createDecipheriv()*, *decipher.setAuthTag()*, *decipher.update()* a *decipher.final()*, jsou zašifrovaná data dešifrována a vrácena v čitelné textové podobě jako původní heslo uživatele, které je následně spolu s uživatelským jménem použito v autentizační hlavičce požadavku na server ISDS. [59]

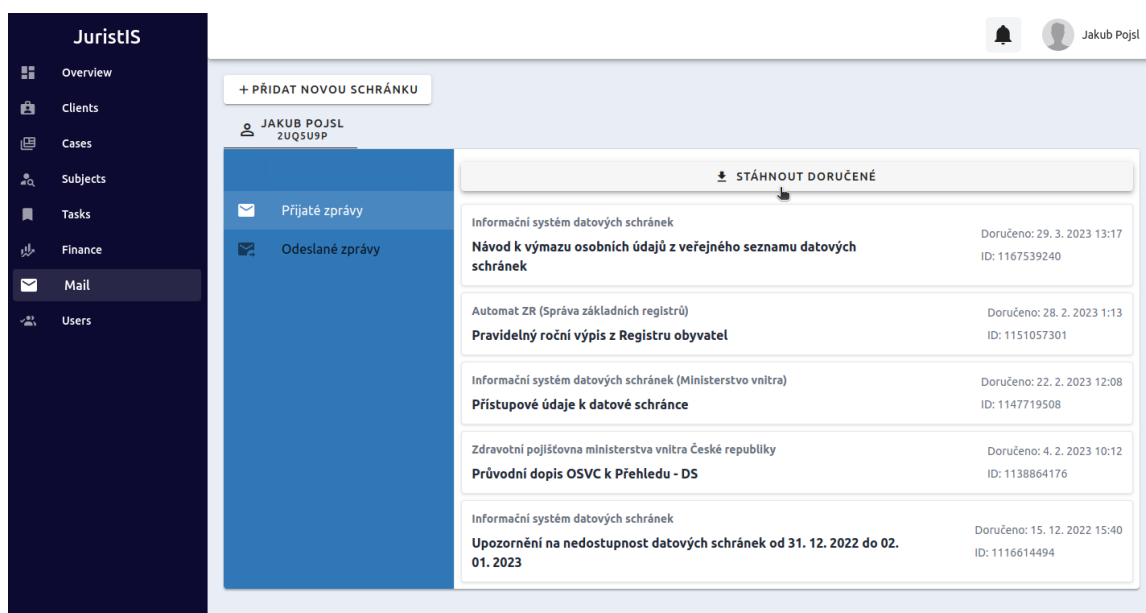
²⁰<https://nodejs.org/api/crypto.html>

5.5.3 Operace se zprávami

V rámci systému bylo implementováno několik základních operací pro přístup ke zprávám z jednotlivých datových schránek uživatele.

Získání seznamu přijatých a odeslaných zpráv

Toto je zásadní operace, jelikož podle dokumentace ISDS se jedná zákoně o jedinou externí operaci přístupnou skrz webové služby, která je považována jako ekvivalentní přihlášení do webového portálu ISDS a tak při jejím vyvolání jsou veškeré zprávy ve schránce považovány za doručené ve smyslu zákona. Vyvolání této operace tedy není nikdy provedeno automaticky a je nutné aby si její provedení vyžádal sám uživatel pomocí tlačítka v uživatelském rozhraní pro stažení doručených zpráv. *API* koncový bod *messages/get-all-received-ds*, tam jsou na základě query parametru id datové schránky z databáze získány informace o datové schránce včetně přihlašovacích údajů uživatele, je ověřeno, že schránka patří přihlášenému uživateli a je dešifrováno přihlašovací heslo. Dále je využito funkce *getListOfReceivedMessages* z modulu *isdsCommunicator*, která vytváří *SOAP* požadavek na ISDS server, spolu s autentizační hlavičkou. Pokud vše proběhne v pořádku, server vrací seznam přijatých zpráv ve schránce uživatele [62]. Výsledek této operace je k vidění na následujícím obrázku 5.12.



Obrázek 5.12: Snímek obrazovky sekce pošty se staženými doručenými zprávami v ukázkové datové schránce

Zobrazení detailu zprávy a její stažení

U každé zprávy je možné zobrazit její detail včetně přiložených souborů, které je možné stáhnout. Pro stažení datové zprávy je vytvořen *API* koncový bod *messages/download*, vyžadující parametry ID datové schránky v databázi a unikátní ID zprávy pro stažení v systému *ISDS*. Při požadavku o stažení zprávy je provedena autentizace uživatele a získání informací o jeho datové schránce stejným způsobem jako u předešlých operací. Poté je využito funkce *downloadMessage* z modulu *isdsCommunicator* pro získání obsahu zprávy

ze serveru *ISDS* na základě jejího ID. Získaná *XML* odpověď je opět převedena do formátu *JSON*, který je následně vrácen jako odpověď na požadavek. Obsahem datové zprávy jsou též přiložené soubory, jejichž obsah je kódován ve formátu *base64*. U souboru je též obsažen jeho mime typ a název, který je zobrazen uživateli a je mu umožněno soubor stáhnout. [62]

5.6 Vykazování a účtování vykonané práce

Možnost vykazování a účtování vykonané práce je klíčovou funkcionalitou systému pro advokáty a advokátní kanceláře. Tato funkcionalita umožňuje advokátům sledovat a fakturovat svou práci pro jednotlivé klienty a případy s nimi řešenými. Díky tomu je zajištěna transparentnost práce a přesné účtování odvedených hodin. Kromě toho je možné snadno analyzovat výkonnost jednotlivých zaměstnanců a optimalizovat jejich nasazení na různé úkoly.

The screenshot shows a web form titled "Přidat nový úkon" (Add new task). It includes a toggle switch for "Podle tarifu" (According to tariff) which is turned on. The form contains several input fields and sections:

- Select Action:** "Úkon podle §11 odst. 1 a"
- Typ aktivity:** "Právní služby"
- Popis aktivity:** "převzetí a příprava zastoupení nebo obhajoby na základě smlouvy o poskytnutí právních služeb"
- Místo a čas:**
 - Začátek: 05/16/2023
 - Začátek: 12:00
 - Konec: 14:00
 - Reálný čas: 2h 0min
 - Počet jednotek: 1
- Související:**
 - Klient: (dropdown)
 - Případ: (dropdown)
 - Jako aktivita
 - Účtovatelné
- Odměna [Kč]:**
 - Tarifní hodnota: 20000
 - Mimosmluvní odměna: 1900
 - Násobek: 1
 - Režijní paušál: 300
 - Celková cena: 2200

At the bottom, there are two buttons: "Zrušit" (Cancel) and "Uložit" (Save).

Obrázek 5.13: Snímek obrazovky okna s formulářem pro přidání nového úkonu s výpočtem odměny podle advokátního tarifu.

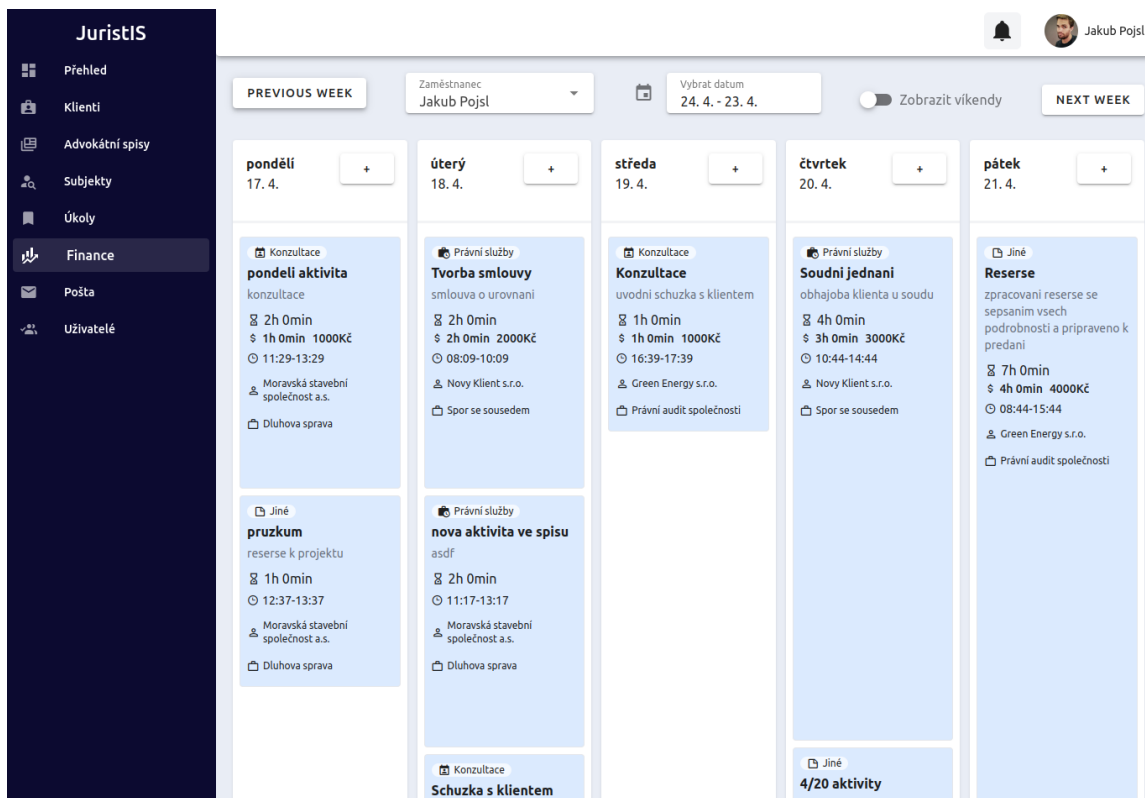
Odměna za některé úkony se může řídit tzv. advokátním tarifem popsaným vyhláškou č. 177/1996 Sb., která stanovuje postupy pro výpočet odměny na základě výše tarifní hodnoty, která typicky vychází z výše peněžitého plnění, ceny věci anebo práva v době započetí úkonu právní služby, jichž se právní služba týká. Dále stanovuje kategorie právních úkonů a další náhrady spojené s poskytováním právních služeb [28]. Pro zjednodušení procesu výpočtu odměny byla vytvořena kalkulačka pro základní výpočet odměny na základě výše tarifní hodnoty, kategorie úkonu a režijního paušálu. Kalkulačka je dostupná přímo v okně pro přidání nového úkonu, jak je vidět na předchozím obrázku 5.13, či v sekci fakturace blíže popsané dále.

5.6.1 Vykazování a přehled práce

Pro **zaznamenávání práce** slouží funkcionalita pro vytváření **úkonů**. Úkony mohou reprezentovat skutečné účtovatelné právní úkony vykonané zaměstnancem, ale i běžné činnosti v rámci jeho práce pro advokátní kancelář. Úkony jsou rozděleny do **4 kategorií** - konzultace, právní služby, analýza a jiné. Každý úkon má kromě názvu a volitelného popisu specifikován začátek a konec, což značí **reálný čas** strávený na úkonu. Vedle toho je možné zaznamenat **účtovaný čas**, který může být klientovi vyúčtován, pokud se jedná o účtovatelný úkon. V takovém případě je zaznamenána i **cena za hodinu** a je vypočítána **celková cena za úkonu**. Alternativně je možné zvolit možnost automatického výpočtu výše odměny podle advokátního tarifu na základě vybraných kritérií. Vykazování práce je možné provádět přímo ze spisu, ze záložky klienta nebo skrze týdenní přehled práce. V případě přidání úkonu u klienta či spisu, je automaticky úkon propojen s tímto klientem či jeho konkrétním spisem. Úkon může být též označen jako aktivita a být tak zařazen do časové osy, pokud se například jedná o významnější úkon.

Pro umožnění souvislého přehledu vlastní provedené práce a efektivnějšího řízení času bylo implementováno rozhraní **týdenního přehledu práce**. Díky tomuto nástroji mohou zaměstnanci snadno sledovat svůj **pracovní výkon** a **provedené úkony** v rámci celého **pracovního týdne**, zaznamenávat vykonanou práci za jednotlivé dny, jednodušeji plánovat své úkoly a případně koordinovat svou činnost s kolegy. Týdenní přehled poskytuje možnost přepínat mezi jednotlivými týdny nebo vybrat konkrétní datum v kalendáři, což umožňuje zaměstnancům flexibilně pracovat s historií svých úkonů a získávat **přehled o svém výkonu** v průběhu času. Při načítání přehledu je automaticky zobrazen přehled pro aktuální týden a pro přihlášeného uživatele. U každého úkonu v přehledu jsou **zobrazeny nejdůležitější informace** včetně případného odkazu na konkrétního klienta nebo spis, ke kterému je vázaný. Tato funkce usnadňuje přístup k stránkám detailu klienta respektive spisu a zjednodušuje tak navigaci mezi různými částmi systému. Pro lepší vizuální představu stráveného času jednotlivými úkony je velikost jednotlivých záznamů úkonů v týdenním přehledu úměrná době celkového reálného času úkonu.

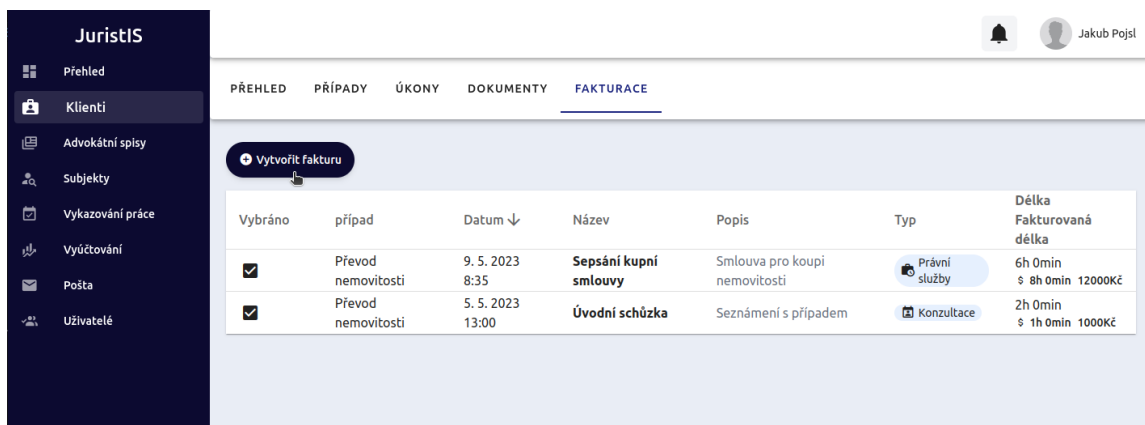
Takovýto přehled je zásadním přínosem i pro **vedoucí pracovníky**, jelikož jim může poskytnout lepší ucelenou představu o **práci celého týmu** i **jednotlivých zaměstnanců**. Proto je umožněno uživatelům s rolí vedoucího zobrazit **přehledy pro jednotlivé zaměstnance**. Tímto způsobem mohou efektivně sledovat a řídit práci celého týmu, což může přispět k **většímu výkonu** a **optimálnějšímu využití času**.



Obrázek 5.14: Snímek obrazovky sekce týdenního přehledu vykonané práce

5.6.2 Tvorba vyúčtování a generování faktur

Pod záložkou vyúčtování jsou pro každého klienta zobrazeny veškeré účtovatelné úkony (k vidím na následujícím obrázku 5.15) provedené zaměstnanci, které ještě nebyly vyúčtovány. Uživatel s odpovídající pravomocí zde může označením vybrat libovolný počet úkonů pro provedení vyúčtování na jméno klienta a automatické vygenerování faktury.



Obrázek 5.15: Snímek obrazovky záložky fakturace ve spisu klienta s vybranými úkony k vyúčtování

Vybrané úkony se následně propíší do formuláře pro tvorbu faktury již jako jednotlivé **vyúčtované položky** s jejich cenou. Uživatel má dále možnost upravit daňovou sazbu pro jednotlivé položky, přejmenovat je či přidat dodatečné vlastní položky k fakturaci. Do formuláře jsou **automaticky vloženy** i informace o advokátní kanceláři jakožto dodavateli a informace o klientovi jakožto odběrateli služby. Tyto informace je případně možné pro účel fakturace upravit. Je možné dále specifikovat číslo faktury, datum vystavení a datum splatnosti. Navíc je přidán i prostor pro případné další doplňující informace pro větší přizpůsobení individuálním potřebám.

Nová Faktura

Číslo faktury: 1 | Vystaveno dne: 09.05.2023 | Splatné do: 31.05.2023

Dodavatel		Zákazník	
Název společnosti Jakub Office		Jméno zákazníka Jan Nový	
Adresa Spojovací 230		Adresa Kotlářská 169	
PSČ 60201	Město Brno	PSČ 60202	Město Brno
ICO 23921232	DIC	ICO 11660295	DIC

Doplňující informace

Položka	Daň	Počet jednotek	Cena za jednotku	Celková cena
Sepsání kupní smlouvy	21	8	1500	14520.00 Kč
Úvodní schůzka	21	1	1000	1210.00 Kč
				15730.00 Kč

+ PŘIDAT POLOŽKU

VYTVOŘIT FAKTURU

Obrázek 5.16: Snímek obrazovky stránky pro tvorbu nové faktury pro klienta s předvyplněnými údaji ze spisu

Veškeré informace z formuláře jsou po odeslání na server využity pro **vygenerování faktury** ke stažení ve formátu PDF a pro tvorbu **záznamu v databázi**. PDF soubor faktury je vytvořen s pomocí npm knihovny *pdf-lib*, která nabízí rozsáhlou funkcionalitu pro vytváření, čtení a úpravu PDF dokumentů přímo v *Node.js* prostředí. Pomocí funkcí knihovny byla vytvořena jednoduchá **šablona** odpovídající běžné struktuře faktur, do které jsou dynamicky vložena konkrétní data. Data výsledného PDF dokumentu jsou poté převedena do formátu BASE64 pro uložení do databáze a umožnění stažení faktury na straně klienta. V databázi je vytvořen záznam faktury spojený s klientem a záznamy pro její jednotlivé položky. Následně je vrácena jako odpověď na stranu klienta, kde je umožněno vygenerovanou fakturu v PDF formátu **stáhnout**. Jednoduchá demonstrace využití této knihovny pro tvorbu PDF je k vidění na následujícím úryvku kódu.

```
async function createPdf() {
  const pdfDoc = await PDFDocument.create();
  const page = pdfDoc.addPage([600, 400]);
  const text = 'Text';

  page.drawText(text, {
    x: (page.getSize().width - textWidth) / 2,
    y: (page.getSize().height - textHeight) / 2,
    size: 10,
  });

  const pdfBytes = await pdfDoc.save();
  return pdfBytes;
}
```

Výpis 5.2: Ukázka kódu pro vytvoření jednoduchého PDF souboru pomocí knihovny pdf-lib

Kapitola 6

Testování a nasazení

V této kapitole je popsán přístup k testování a jsou shrnuty jeho výsledky. Otestování systému na reálných datech a případech použití bylo klíčovou součástí vývoje pro celkové ověření očekávané funkcionality a zhodnocení implementace, získání zpětné vazby od potenciálních uživatelů a pro přípravu systému na případné spuštění do reálného provozu. Proces byl rozdělen na testování funkcionality a uživatelské testování. Obě části testování byly provedeny na několika odlišných zařízeních a v různých webových prohlížečích, pro ověření správného zobrazení a celkové kompatibility. Dále je zde blíže přiblížen postup při nasazení systému na veřejně přístupný testovací server, které umožnilo jednodušší demonstraci funkcionality a uživatelské testování na libovolném zařízení. Následně je nastíněn průběh uživatelského testování v konkrétní advokátní kanceláři i s několika zaměstnanci odlišných advokátních kanceláří.

6.1 Testování funkcionality

Během testování funkcionality byl systém podroben zkoumání pro ověření, zda všechny klíčové prvky fungují podle očekávání a splňují požadavky. Testování se zaměřovalo jak na komunikaci s externími systémy a získávání dat z veřejných databází tak na ověření obecné funkcionality jako práce s dokumenty, správa klientů a spisů, vykazování a účtování práce, a další z hlavních funkcí. Testování probíhalo na základě předem připravených testovacích scénářů, které vycházely ze specifikovaných případů užití v části 4.1.3 a bylo prováděno převážně ručně. Pro některé testovací scénáře a vybrané funkce byly vytvořeny automatické testy, které jsou dostupné v adresáři *tests*. Pro tvorbu automatických testů bylo využito *npm* knihovny *jest*¹, která umožňuje efektivní tvorbu testů díky funkcionalitě pro automatické sledování změn, mockování funkcí, aserce a další.

Jako příklady testů je možné uvést např. následující testovací scénáře:

Testování funkce pro správu spisů

V tomto testu byl ověřován proces vytváření, úpravy a odstranění spisu v rámci systému. Testovací scénář zahrnoval následující kroky:

1. Vytvoření nového spisu s platnými údaji a přiřazením klienta.

¹<https://jestjs.io/>

2. Ověření, že nový spis se zobrazuje v seznamu spisů a je správně přiřazen k danému klientovi.
3. Úprava informací o spisu a ověření, že změny byly úspěšně uloženy.
4. Odstranění spisu ze systému a ověření, že byl úspěšně odstraněn ze seznamu spisů.

Testování funkcí pro práci s dokumenty

V tomto testu byl ověřován proces nahrávání, úpravy a odstranění dokumentů a jejich verzí v rámci spisů a související funkcionality pro práci s implementovaným souborovým systémem

Testovací scénář zahrnoval následující kroky:

1. Nahrání nového dokumentu do složky spisu.
2. Ověření, že dokument byl úspěšně nahrán a je viditelný v rámci složky spisu.
3. Přidání nové verze dokumentu a ověření úspěšného nahrání a propojení s původním dokumentem.
4. Přesunutí dokumentu do vnořené složky a ověření uložení.
5. Opětovné stažení dokumentu i nové verze a ověření jejich obsahu.
6. Odstranění dokumentu ze spisu a ověření, že byl úspěšně odstraněn.

Testování získání dat z infoSoudu

Cílem tohoto testu bylo ověření funkčnosti komunikace se systémem infoSoud pro získávání informací o soudních řízeních a nařízených soudních jednáních a následné správné zobrazení získaných informací. Bylo žádoucí také otestovat funkčnost upozornění při nastalé změně. Při testování byla získávána data o reálně probíhajících řízeních.

Testování se skládalo z následujících kroků:

1. Vyplnění formuláře pro přidání nové spisové značky, který obsahuje pole pro krajský soud, okresní soudu a kód spisové značky.
2. Ověření, že došlo k extrahování informací a zda byly úspěšně uloženy do databáze.
3. Ověření vytvoření aktivit v časové ose a kontrola zda se zobrazené informace shodují s obsahem na stránkách infoSoudu.
4. Manuální odstranění poslední události řízení z databáze, pro simulaci vzniku nové události na straně infoSoudu.
5. Vyvolání kontroly změn řízení.
6. Kontrola zobrazení nového upozornění, uložení “nové” události a zobrazení v časové ose.

Obdobným způsobem bylo provedeno testování i zbývající klíčové funkcionality a vedlo k objevení několika nedostatků, které mohly být následně opraveny.

6.2 Zveřejnění implementovaného systému

Pro zveřejnění implementovaného systému bylo využito **cloudové platformy** Vercel², specializující se na hosting a deployment webových aplikací. Platforma poskytuje možnost jednoduchého nastavení a nabízí řadu užitečných funkcí. Mezi ty patří např. možnost continuous deployment/integration díky propojení s repozitářem projektu ve verzovacím systému, efektivní management odlišných prostředí (produkční/testovací), historie deploymentů, či automatické zřizování *SSL certifikátu*. Výhodou je také, že platforma nabízí limitovaný bezplatný balíček služeb, který umožňuje hosting vlastních projektů zdarma.

Po vytvoření účtu na platformě a založení nového projektu byla provedena konfigurace proměnných prostředí obsahujících citlivé údaje, jako jsou tajné API klíče pro externí služby, či klíč použitý pro šifrování citlivých dat. Následně byl projekt propojen s GitHub³ repozitářem, který byl využíván jako verzovací systém v rámci vývoje celého projektu a obsahuje veškerý kód implementovaného systému. Integrace na platformě Vercel byla následně nastavena tak, že s každým provedením push do hlavní větve připojeného GitHub repozitáře je spuštěn automatický deployment nové verze, která je následně téměř okamžitě přístupná pod specifikovanou url adresou. Toto umožňuje rychlé iterování a jednoduché zveřejnění všech nových změn v kódu.

6.3 Uživatelské testování

Pro ověření funkcionality systému s **reálnými uživateli** a získání zpětné vazby k uživatelskému rozhraní bylo naplánováno uživatelské testování. Testování proběhlo ve spolupráci jak s vybranou advokátní kanceláří v Prachaticích, tak navíc zahrnovalo i několik zaměstnanců z dvou různých advokátních kanceláří v Brně, aby byla zajištěna získání širší škály názorů a zkušeností. Konkrétně se jednalo o **menší advokátní kanceláře** (<10 zaměstnanců), které v době testování nevyužívali žádný specializovaný interní systém. Je důležité podotknout, že zpětná vazba byla díky konzultacím získávána již v průběhu návrhu a implementace systému a některé úpravy na základě doporučení tak již byly do systému rovnou zapracovány. Patřilo mezi ně například možnost přidání několika datových schránek, odkázání na spis či klienta v rámci aktivit v časové ose či možnost zobrazení víkendů v přehledu práce a další menší doporučení mimo jiné i z pohledu vzhledu uživatelského rozhraní.

Pro uživatelské testování byly připraveny **testovací scénáře**, které simulovaly běžné pracovní úkoly, jež uživatelé v systému mohou vykonávat. Tyto scénáře pokrývaly vybrané **klíčové funkce**, mezi které patřila práce s dokumenty, správa klientů a spisů, získávání informací z externích zdrojů a vykazování a účtování práce. Obecně vypadaly testovací scénáře podobně jako ukázky testovacích příkladů v předešlé podkapitole 6.1. Větší důraz byl ale kladen na **pochopení funkcionality** uživatelem a ověření zda výstupy odpovídaly jejich očekáváním. Účastníci testování byli seznámeni s těmito scénáři a byli vyzváni k jejich provedení v rámci testovacího prostředí. Pro zhodnocení výsledků a získání zpětné vazby byl sestaven **dotazník** pro subjektivní zhodnocení jednotlivých funkcí a uživatelského rozhraní ve spojení s jejich užitečností a efektivitou z pohledu běžného fungování právní praxe. To umožnilo získat lepší přehled o kvalitách a slabších místech systému očima uživatelů. Výsledkem uživatelského testování byly též hodnotné poznatky pro specifikování možných budoucích rozšíření systému a identifikaci potenciálně užitečných rozšiřujících funkcí, které jsou blíže rozebrány v závěrečné sekci 7.2.

²<https://vercel.com/>

³<https://github.com/>

Kapitola 7

Závěr

Tato závěrečná kapitola práce je věnována celkové rekapitulaci práce a shrnutí dosažených výsledků. Jsou zde vyzdvíženy nejdůležitější implementované funkce a přínosy vytvořeného systému a jsou zmíněny odlišnosti od existujících systémů. Na závěr jsou nastíněny možnosti budoucích rozšíření.

7.1 Shrnutí dosažených výsledků

Cílem práce bylo navrhnout a implementovat **systém pro advokátní kanceláře** s agregací dat z **veřejných databází**. Systém by měl advokátním kancelářím pomoci s automatizací a zefektivněním procesů a přinést navíc funkce pro správu spisů, dokumentů a evidenci pošty. Požadavkem bylo též zahrnout funkce pro generování podkladů pro účtování provedené práce klientům.

Výsledkem práce je **kompletní implementovaný systém** s rozsáhlou funkcionalitou umožňující **zefektivnění agendy advokátní kanceláře**. Systém byl implementován na základě vlastní podrobné analýzy procesů advokátní kanceláře a jejich práce s veřejnými rejstříky a na základě poznatků získaných během průzkumu existujících řešení. Řešení je koncipováno jako **webová aplikace**, což zajišťuje dostupnost nezávisle na platformě a je přístupné z webového prohlížeče. Díky **responzivnímu designu** je aplikace navíc přizpůsobena pro širokou škálu zařízení s libovolnou velikostí displeje, včetně **mobilních zařízení**.

Výsledný systém umožňuje uživatelům efektivně spravovat informace o klientech a případech s nimi řešenými a nabízí plnou podporu pro **digitální vedení spisů klienta**. Pomocí automatického získávání dat z **veřejných databází** je umožněn jednoduchý přístup k užitečným informacím a **automatizace vybraných procesů**. Díky informacím z infoSoud je možné sledovat **průběh soudních řízení** přímo ze spisu klienta a automaticky **upozorňovat uživatele** na nastalé změny. Je tak eliminována nutnost pravidelného manuálního kontrolování. Agregací dat z **obchodního a živnostenského rejstříku** je uživateli nabídnut jednoduchý a přehledný přístup k aktuálním informacím o subjektech přímo v systému a umožňuje **automatické předvyplnění** vybraných údajů při různých činnostech. Je tak ušetřen čas oproti manuálnímu zadávání veškerých údajů. Pro přístup k agregovaným datům z veřejných databází bylo též zveřejněno externí API rozhraní, umožňující jednoduše získat vybraná data z koncových databází v jednotném formátu JSON třetími stranami.

Díky implementaci **uživatelských rolí a autorizace**, je možné definovat pravomoce jednotlivých uživatelů a zpřístupnit tak zaměstnancům advokátní kanceláře požadovanou

funkcionalitu a vybrané části systému. Pro evidenci, **správu a sdílení dokumentů** bylo implementováno vlastní řešení **souborového systému**, který je provázaný s jednotlivými klienty a spisy. Souborový systém podporuje kromě běžných operací také **správu verzí dokumentů** a možnost **připojení souborů z externích úložišť** pomocí odkazu. Pro zefektivnění správy pošty byla provedena integrace se **systémem datových zpráv** a implementována funkcionalita pro evidenci další **libovolné korespondence**.

Pro podporu **účtování** provedené práce klientům systém umožňuje zaměstnancům zaznamenávat **úkony** vykonané v rámci spisu či klienta. Na základě úkonů je možné **vygenerovat fakturu** ve formátu PDF, opět s automatickým **předvyplněním** potřebných údajů. Úkony slouží zároveň pro **vykazování** provedené práce jednotlivými zaměstnanci a mohou také sloužit vedoucím pracovníkům pro lepší **přehled nad vytížeností** zaměstnanců a umožnit **efektivnější plánování**. Pro celistvý pohled na vykonanou práci v průběhu času byl přidán týdenní přehled úkonů.

Implementovaný systém **splňuje specifikované požadavky** a některé části byly přidány nad rámec původního návrhu díky zpětné vazbě získané na konzultacích se zaměstnanci advokátní kanceláře. Uživatelským testováním bylo potvrzeno, že systém má **potenciál zefektivnit** fungování advokátní kanceláře, která zatím žádný systém nevyužívá a implementovaná funkcionalita byla shledaná **celkově nápomocnou** osobami z právní praxe. Je zde tak otevřená možnost pro navázání potenciální spolupráce s vybranou advokátní kanceláří pro doladění a nasazení systému v reálném provozu.

Nabízenou funkcionalitou se vytvořený systém ve vybraných ohledech blíží existujícím profesionálním řešením a některé z nich i přesahuje například svým více propracovaným souborovým systémem provázaným s klienty a spisy a možným přiložením souborů z online úložišť. Jedním z rozšíření existujících služeb a databází je např. **zveřejněné API rozhraní** nabízející efektivní přístup k vybraným datům v jednotném formátu JSON. Jako výhoda může být též viděno přehledné **uživatelské rozhraní** optimalizované i pro **mobilní zařízení**, což některé ze systémů nenabízejí. Některé z existujících systémů však nabízejí pokročilejší funkcionalitu a více možností, jejichž implementace by přesahovala rámec této práce. Je zde tak stále prostor pro zlepšování a rozšiřování systému o další funkce a některé možnosti jsou přiblíženy v příští podkapitole.

Vzhledem k celkové pozitivní zpětné vazbě získané od zaměstnanců advokátní kanceláře věřím, že systém má potenciál reálného využití a mám tak v plánu pokračovat v jeho vývoji s cílem nasazení do reálného provozu. V době zveřejnění práce je systém zveřejněn pod url adresou <https://www.juristis.cz/>. Na stejné adrese je též dostupné zmiňované veřejné API rozhraní.

7.2 Možná budoucí rozšíření

V následujících závěrečných odstavcích jsou stručně nastíněny některé možnosti rozšíření a budoucího využití vytvořeného systému. Možnosti rozšíření byly identifikované v průběhu implementace, na základě analýzy existujících systémů a zpětné vazby od zaměstnanců advokátní kanceláře.

Mohla by mezi ně patřit například následující funkcionalita:

1. **Integrace s dalšími veřejnými databázemi a službami** pro rozšíření informací dostupných přímo v systému a zefektivnění dalších procesů. Mohlo by se jednat například o katastr nemovitostí či elektronické platební rozkazy.

2. **Propojení s e-mailovou schránou** by mohlo být užitečnou integrací pro možnost správy veškeré pošty na jednom místě a automatické provázání korespondence s klienty a spisy. Zahrnuty by mohly být i funkce pro tvorbu šablon emailů a nastavení automatických odpovědí.
3. **Podpora pro elektronické podepisování dokumentů** by mohlo zefektivnit práci s dokumenty vyžadující autentizaci pomocí certifikátu. Související funkcionalitou by mohla být autorizovaná konverze dokumentů z fyzické do digitální podoby a obráceně.
4. **Funkce pro spolupráci a komunikaci mezi uživateli** pro usnadnění práce v týmech a zlepšení koordinace mezi zaměstnanci. Příkladem by mohla být možnost vytváření a sdílení úkolů, kalendář s termíny, chatovací platforma pro efektivní komunikaci mezi kolegy, nebo možnost komentování u klientů či spisů.
5. **Funkce pro tvorbu šablon dokumentů a smluv** by mohly nabídnout možnost automatického předvyplnění informací o klientovi na vybraná místa v šabloně dokumentu což by mohlo zjednodušit proces vytváření nových smluv.

Uvedený výčet se zaměřuje pouze na zajímavější rozšíření, která by mohla být dále rozvíjena. Vzhledem k rozsáhlosti problematiky a počtu procesů, které je třeba v rámci právní agendy řešit, je možné vytvořit další optimalizační funkce, které by mohly přispět ke zvýšení efektivity.

Literatura

- [1] *About Node.js* [online]. [cit. 2023-05-13]. Dostupné z: <https://nodejs.org/en/about>.
- [2] *About PostgreSQL* [online]. [cit. 2023-05-10]. Dostupné z: <https://www.postgresql.org/about/>.
- [3] *Administrativní registr ekonomických subjektů* [online]. [cit. 2023-05-09]. Dostupné z: <https://wwwinfo.mfcr.cz/ares/>.
- [4] *Aplikace ePodatelna* [online]. [cit. 2023-05-09]. Dostupné z: <https://epodatelna.justice.cz/ePodatelna/homepage>.
- [5] *Architektura systému Praetor* [online]. [cit. 2023-05-14]. Dostupné z: <https://www.wolterskluwer.com/cs-cz/solutions/praetor/architektura>.
- [6] *Beck-online* [online]. [cit. 2023-05-09]. Dostupné z: <https://www.beck-online.cz/>.
- [7] *Codexis* [online]. [cit. 2023-05-09]. Dostupné z: <https://codexis.cz/>.
- [8] *Databáze rozhodnutí NALUS* [online]. [cit. 2023-05-09]. Dostupné z: <https://www.usoud.cz/vyhledavani-rozhodnuti-us>.
- [9] *Databáze vybraných rozhodnutí Evropského soudu pro lidská práva* [online]. [cit. 2023-05-09]. Dostupné z: <http://eslp.justice.cz/>.
- [10] *Dálkový přístup k údajům katastru nemovitostí České republiky* [online]. [cit. 2023-05-09]. Dostupné z: <https://www.cuzk.cz/Aplikace-DP-do-KN/Aplikace-DP-do-KN/Dalkovy-pristup-do-KN.aspx>.
- [11] *Evidence soudních rozhodnutí vrchních a krajských soudů* [online]. [cit. 2023-05-09]. Dostupné z: https://www.nsoud.cz/judikaturavks/judikatura_vks.nsf/uvod.
- [12] *Evolio* [online]. [cit. 2023-05-14]. Dostupné z: <https://evolio.cz/>.
- [13] *InfoJednání* [online]. [cit. 2023-05-09]. Dostupné z: <https://portal.justice.cz/ejustice/info-jednani.html>.
- [14] *InfoSoud* [online]. [cit. 2023-05-09]. Dostupné z: <http://portal.justice.cz/ejustice/info-soud.html>.
- [15] *Insolvenční rejstřík* [online]. [cit. 2023-05-14]. Dostupné z: <https://isir.justice.cz/isir/common/index.do>.
- [16] *Inteligentní spis* [online]. [cit. 2023-05-14]. Dostupné z: <https://ispis.cz/>.

- [17] *Isak* [online]. [cit. 2023-05-14]. Dostupné z: <https://www.isak.cz/>.
- [18] *Judikatura Nejvyššího soudu ČR* [online]. [cit. 2023-05-09]. Dostupné z: https://www.nsoud.cz/Judikatura/ns_web.nsf/WebSpreadSearch/.
- [19] *Nahlizení do katastru nemovitostí* [online]. [cit. 2023-05-08]. Dostupné z: <https://nahlizenedokn.cuzk.cz/>.
- [20] *Nuxt 3* [online]. [cit. 2023-05-13]. Dostupné z: <https://nuxt.com/>.
- [21] *Portál justice.cz* [online]. [cit. 2023-05-09]. Dostupné z: <https://justice.cz/>.
- [22] *Portál živnostenského podnikání* [online]. [cit. 2023-05-06]. Dostupné z: <https://www.rzp.cz/verejne-udaje/napoveda/>.
- [23] *Praetor* [online]. [cit. 2023-05-14]. Dostupné z: <https://www.wolterskluwer.com/cs-cz/solutions/praetor>.
- [24] *Prisma* [online]. [cit. 2023-05-13]. Dostupné z: <https://www.prisma.io/>.
- [25] *Supabase* [online]. [cit. 2023-05-10]. Dostupné z: <https://supabase.com/>.
- [26] *Vue.js* [online]. [cit. 2023-05-13]. Dostupné z: <https://vuejs.org/>.
- [27] *Vyhledávač rozhodnutí Nejvyššího správního soudu* [online]. [cit. 2023-05-09]. Dostupné z: <https://vyhledavac.nssoud.cz/>.
- [28] *Vyhláška č. 177/1996 Sb., vyhláška o odměnách advokátů a náhradách advokátů za poskytování právních služeb (advokátní tarif)* [online]. [cit. 2023-05-13]. Dostupné z: <https://www.zakonyprolidi.cz/cs/1996-177>.
- [29] *What is Three-Tier Architecture* [online]. [cit. 2023-05-14]. Dostupné z: <https://www.ibm.com/topics/three-tier-architecture>.
- [30] *Wolters Kluwer - ASPI* [online]. [cit. 2023-05-09]. Dostupné z: <https://www.aspi.cz/>.
- [31] *Zákon č. 106/1999 Sb., zákon o svobodném přístupu k informacím* [online]. [cit. 2023-05-09]. Dostupné z: <https://www.zakonyprolidi.cz/cs/1999-106>.
- [32] *Zákon č. 182/2006 Sb., zákon o úpadku a způsobech jeho řešení* [online]. [cit. 2023-05-04]. Dostupné z: <https://www.zakonyprolidi.cz/cs/2006-182>.
- [33] *Zákon č. 256/2013 Sb., zákon o katastru nemovitostí* [online]. [cit. 2023-05-07]. Dostupné z: <https://www.zakonyprolidi.cz/cs/2013-256>.
- [34] *Zákon č. 300/2008 Sb., zákon o elektronických úkonech a autorizované konverzi dokumentů* [online]. [cit. 2023-05-09]. Dostupné z: <https://www.zakonyprolidi.cz/cs/2008-300>.
- [35] *Zákon č. 359/1992 Sb., zákon o zeměměřičských a katastrálních orgánech* [online]. [cit. 2023-05-07]. Dostupné z: <https://www.zakonyprolidi.cz/cs/1992-359>.
- [36] *Zákon č. 85/1996 Sb., zákon o advokacii* [online]. [cit. 2023-05-07]. Dostupné z: <https://www.zakonyprolidi.cz/cs/1996-85>.

- [37] *Zákon č. 89/2012 Sb., občanský zákoník* [online]. [cit. 2023-05-04]. Dostupné z: <https://www.zakonyprolidi.cz/cs/2012-89>.
- [38] *Zákon č. 90/2012 Sb., zákon o obchodních společnostech a družstvech* [online]. [cit. 2023-05-04]. Dostupné z: <https://www.zakonyprolidi.cz/cs/2012-90>.
- [39] *Úřady v ČR* [online]. [cit. 2023-05-09]. Dostupné z: <https://www.statnisprava.cz/rstsp/redakce.nsf/i/home>.
- [40] *Živnostenský rejstřík* [online]. [cit. 2023-05-06]. Dostupné z: https://www.rzp.cz/cgi-bin/aps_cacheWEB.sh?VSS_SERV=ZVWSBJFND.
- [41] *Zákon č. 304/2013 Sb., zákon o veřejných rejstřících právnických a fyzických osob* [online]. 2013 [cit. 2023-05-14]. Dostupné z: <https://www.zakonyprolidi.cz/cs/2013-304>.
- [42] AALST, W. M. van der a STAHL, C. *Modeling Business Processes*. The MIT Press, 2011. ISBN 9780262015387.
- [43] ARLOW, J. *UML a unifikovaný proces vývoje aplikací*. Brno: Computer Press, 2003. ISBN 978-80-7400-659-380-7226-947-X.
- [44] BAUDYŠ, P. *Katastrální zákon*. Praha: C. H. Beck, 2014. ISBN 978-80-7400-525-1.
- [45] BENÁK, J. a ZAHUMENSKÝ, D. *Jak na spolkový rejstřík*. Grada publishing, 2015. ISBN 978-80-247-5672-1.
- [46] BERSON, A. *Client-server Architecture*. McGraw-Hill, 1992. ISBN 0070050767.
- [47] BEČVÁŘ, O., GRUNDOVÁ, L., KREJČÍ, K. a KVAPIL, M. *Zákon o veřejných rejstřících právnických a fyzických osob a o evidenci svěřenských fondů - judikatorní komentář*. CODEXIS publishing, 2021.
- [48] BOODHOO, J.-P. *Design Patterns - Model View Presenter* [online]. 2019 [cit. 2023-05-04]. Dostupné z: <https://learn.microsoft.com/en-us/archive/msdn-magazine/2006/august/design-patterns-model-view-presenter>.
- [49] GOOGLE. *Google Drive API* [online]. 2023 [cit. 2023-05-02]. Dostupné z: <https://developers.google.com/drive/api/guides/about-sdk>.
- [50] HARRIS, C. *Microservices vs. monolithic architecture* [online]. [cit. 2023-05-14]. Dostupné z: <https://www.atlassian.com/microservices/microservices-architecture/microservices-vs-monolith>.
- [51] HRONEK, J. *Informační systémy* [online]. [cit. 2023-05-14]. Dostupné z: <https://phoenix.inf.upol.cz/esf/ucebni/infoSys.pdf/>.
- [52] *Automatické sledování insolvenčního rejstříku* [online]. 2022 [cit. 2023-05-04]. Dostupné z: <https://isir.justice.cz/isir/common/stat.do?kodStranky=SLEDOVANIWS>.
- [53] KIMBLE, C. *Types of Information System and the Classic Pyramid Model* [online]. [cit. 2023-05-14]. Dostupné z: http://www.chris-kimble.com/Courses/World_Med_MBA/Types-of-Information-System.html.

- [54] MARTIN, R. C. *Clean Architecture: A Craftsman's Guide to Software Structure and Design*. Pearson, 2017. ISBN 9780134494272.
- [55] MARŠÍKOVÁ, J. *Insolvenční řízení z pohledu dlužníka a věřitele: příručka zejména pro neprávniky*. Linde Praha, 2014. ISBN 978-80-7201-939-7.
- [56] MOLNÁR, Z. *Podnikové informační systémy*. České vysoké učení technické, 2009. ISBN 9788001043806.
- [57] CONTRIBUTORS, M. *Document Object Model* [online]. 2023 [cit. 2023-05-10]. Dostupné z: https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model/Introduction.
- [58] MSGRAPHDOCSVTEAM. *OneDrive file storage API* [online]. 2022 [cit. 2023-05-02]. Dostupné z: <https://learn.microsoft.com/en-us/graph/onedrive-concept-overview>.
- [59] NAKOV, S. *Practical Cryptography for Developers*. Self-Published, 2018. ISBN 978-619-00-0870-5.
- [60] KUSHWAHA, N. *Best practices for storing passwords securely in a database* [online]. 2022 [cit. 2023-05-01]. Dostupné z: <https://www.learncsdesign.com/best-practices-for-storing-passwords-securely-in-a-database/>.
- [61] SVEJKOVSKÝ, J. *Advokátní právo*. C. H. Beck, 2017. ISBN 978-80-7400-659-3.
- [62] ČR, M. vnitra. *Provozní řád ISDS* [online]. 2022 [cit. 2023-04-24]. Dostupné z: https://info.mojedatovaschranka.cz/info/files/2104_Provozni_rad_ISDS.pdf.