



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ

DEPARTMENT OF INFORMATION SYSTEMS

INFORMAČNÍ SYSTÉM PRO ADVOKÁTNÍ KANCELÁŘ

INFORMATION SYSTEM FOR AN ATTORNEY'S OFFICE

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

RADOVAN BARTÁNUS

VEDOUcí PRÁCE

SUPERVISOR

RNDr. MAREK RYCHLÝ, Ph.D.

BRNO 2017

Zadání bakalářské práce

Řešitel: **Bartánus Radovan**

Obor: Informační technologie

Téma: **Informační systém pro advokátní kancelář**
Information System for an Attorney's Office

Kategorie: Informační systémy

Pokyny:

1. Seznamte se požadavky na software pro evidenci dokumentů a činností v advokátní kanceláři působící ve Slovenské republice. Soustředte se na legislativní a uživatelské požadavky, bezpečnost, přizpůsobitelnost a rozšiřitelnost systému.
2. Dle požadavků v předchozím kroku navrhnete informační systém pro potřeby slovenské advokátní kanceláře. Systém bude umožňovat evidenci dokumentů a pracovních úkonů (např. zadání práce, napojení na související dokumenty, atp.).
3. Po konzultaci s vedoucím navržený systém implementujte, jako aplikaci s webovým uživatelským rozhraním.
4. Výsledný systém důkladně otestujte a, pokud možno, nasadte v praxi.
5. Provedte zhodnocení výsledků práce a diskutujte možná rozšíření.

Literatura:

- Jiří Janda. *Spisová služba advokátní kanceláře s napojením na veřejné databáze*. Diplomová práce, FIT VUT v Brně, 2014. [<http://www.fit.vutbr.cz/study/DP/DP.php?id=15081>]
- Jakub Trefilík. *Správa spisů advokátní kanceláře s napojením na veřejné databáze*. Bakalářská práce, FIT VUT v Brně, 2013. [<http://www.fit.vutbr.cz/study/DP/BP.php?id=15080>]
- Jiří Barbořík. *Informační systém pro řízení toku dokumentů*. Diplomová práce, FIT VUT v Brně, 2009. [<http://www.fit.vutbr.cz/study/DP/DP.php?id=8597>]

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 a 2.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

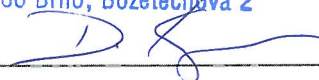
Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Rychlý Marek, RNDr., Ph.D.**, UIFS FIT VUT

Datum zadání: 1. listopadu 2016

Datum odevzdání: 17. května 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav informačních systémů
612 66 Brno, Božetěchova 2


doc. Dr. Ing. Dušan Kolář
vedoucí ústavu

Abstrakt

Táto práca sa zaoberá výpočetnými nástrojmi, určenými na zvýšenie efektivity, zníženie náročnosti a monitorovanie pracovných procesov uskutočňovaných v advokátskych kanceláriách na území Slovenskej republiky. Zároveň analyzuje súčasne dostupné software-ové prostriedky s podobným účelom. Taktiež preveruje bezpečnostné požiadavky, určené zákonmi Slovenskej republiky, zaisťujúce ochranu osobných údajov zákazníkov a pracovníkov advokátskych kancelárií. Výsledky analýzy napomohli k návrhu a implementácii výslednej webovej aplikácie slúžiacej ako informačný systém pre advokátske kancelárie. Aplikácia je zároveň rozšírená o vyhľadávanie záznamov v obchodnom registri Slovenskej republiky a monitorovanie ich zmien.

Abstract

This thesis covers the computational tools designated for the purposes of increasing efficiency, lowering difficulty and monitoring of work processes in the attorney offices in Slovak republic. It also analyses the current available software tools with a similar purpose. At the same time it examines safety requirements, specified by the laws of Slovak republic, which ensure the protection of personal data of clients and employees of attorney offices. The results of the analysis helped designing and implementing the final web application, which can be used as an information system for attorney offices. The final application includes a search for records in the register of companies of Slovak republic, and notifying users of any changes in the records.

Kľúčové slová

advokátska kancelária, databáza, MySQL, entitno-relačný diagram, diagram prípadov užitia, sekvenčný diagram, PHP, Symfony, Doctrine, verejná databáza, organizácia práce, webová aplikácia

Keywords

attorney office, database, MySQL, entity-relationship diagram, use case diagram, sequence diagram, PHP, Symfony, Doctrine, public database, workforce management, web application

Citácia

BARTÁNUS, Radovan. *Informační systém pro advokátní kancelář*. Brno, 2017. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce RNDr. Marek Rychlý, Ph.D.

Informační systém pro advokátní kancelář

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána RNDr. Marka Rychlého, Ph.D.. Ďalšie informácie mi poskytli JUDr. Ján Bartánus a JUDr. Miloslav Hricko. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....
Radovan Bartánus
17. mája 2017

Podakovanie

Velká vdaka patrí pánovi RNDr. Markovi Rychlému, Ph.D., za vedenie a pomoc pri tvorbe bakalárskej práce a poskytnutú odporúčanú literatúru. Za inšpiráciu, právne rady a priblíženie problematiky advokátskej praxe ďakujem pánom JUDr. Jánovi Bartánusovi a JUDr. Miloslavovi Hrickovi.

Obsah

1	Úvod	3
1.1	Štruktúra práce	3
2	Legislatívne a administratívne požiadavky	5
2.1	Legislatívne požiadavky	5
2.1.1	Evidencia informačného systému	5
2.1.2	Bezpečnostné požiadavky	7
2.2	Procesy v advokátskej kancelárii	8
2.3	Obchodný register	9
2.4	Záver	10
3	Existujúce software-ové riešenia	11
3.1	LEGUS	11
3.2	AdvocatusDigital	12
3.3	Tribunal	13
3.4	ADVOS	13
3.5	Zhrnutie	14
4	Analýza požiadavok na navrhovaný systém	15
4.1	Správa spisov advokátskej kancelárie	15
4.2	Napojenie na obchodný register	16
4.3	Pripomienky o udalostiach	16
4.4	Správa zamestnancov advokátskej kancelárie	16
4.5	Nástroje	17
4.6	Bezpečnosť	17
4.7	Záver	17
5	Návrh systému	18
5.1	Diagram prípadov použitia	18
5.2	Entitno-relačný diagram	19
5.3	Diagram interakcie so systémom	20
5.4	Záver	20
6	Implementácia systému	24
6.1	Štruktúra programu	24
6.2	Konfigurácia	26
6.2.1	Príkazy cron	26
6.2.2	Inicializácia systému	27

6.3	Webové užívateľské rozhranie systému	27
6.3.1	Navigácia	27
6.3.2	Zobrazenia zoznamov údajov	28
6.3.3	Detaily	29
6.3.4	Formuláre	31
6.4	Databázový systém	33
6.4.1	Štruktúra databázy	33
6.4.2	Práca s databázou	34
6.5	Napojenie na obchodný register	35
6.5.1	Vyhľadanie záznamu	35
6.5.2	Kontrola zmien záznamov	35
6.6	Súborový systém	36
6.6.1	Dokumenty	36
6.6.2	Záloha	36
6.7	Testovanie	36
6.7.1	Jednotkové testovanie	37
6.7.2	Testovanie funkcionality	37
6.8	Zhrnutie	37
7	Záver	38
	Literatúra	39
A	Obsah priloženého média	41
B	Ukážka užívateľského rozhrania	42

Kapitola 1

Úvod

Informačné technológie sa v súčasnej dobe využívajú v čoraz širšej škále odvetví na mnoho účelov. Napriek tomu, že vo väčšine prípadov dokážu zefektívniť a automatizovať denne vykonávané procesy, sa v niektorých odvetviach neustále používajú zastaralé postupy. Jedným z týchto odvetví je aj advokátska prax.

V niektorých advokátskych kanceláriách na území Slovenskej republiky sa ešte doteraz používajú výlučne papierové adresáre a organizácia práce je v mnohých prípadoch založená len na slovnej dohode. Tento archaický prístup sa ale s novou vlnou advokátov pomaly mení a pristupuje sa k technologicky modernejším riešeniam.

Práve preto sa táto bakalárska práca zaoberá možnosťou zjednodušenia práce v advokátskej kancelárii na území Slovenskej republiky použitím informačných technológií. Skúma problémy pracovných záležitostí v advokátskej kancelárii a prejednáva možnosť ich automatizácie. Ďalej analyzuje dostupné moderné komerčné prostriedky využívané v advokátskych kanceláriách. Zohľadňuje taktiež problémy s bezpečnosťou a ochranou osobných údajov.

Produktom tejto bakalárskej práce je informačný systém s webovým rozhraním, použiteľný na elektronickú správu zamestnancov, spisov, dokumentov a udalostí. Využívanie tohto systému má potenciál zefektívniť organizáciu práce v advokátskej kancelárii, zjednodušiť správu a vyhľadávanie dokumentov a upozorňovať užívateľov na nové, či blížiac sa pracovné udalosti.

V rámci informačného systému môže užívateľ vyhľadávať záznamy vo verejnej databáze Obchodného registra Slovenskej republiky¹ a následne byť upozornený na ich zmeny.

1.1 Štruktúra práce

Štruktúra tejto práce vychádza z mnou vykonávaného priebežného výskumu, návrhu a následnej implementácii informačného systému.

V druhej kapitole opisujem základné požiadavky určené bežnými operáciami uskutočňovanými v advokátskych kanceláriách na Slovensku. Taktiež uvádzam požiadavky vyplývajúce zo zákonov Slovenskej republiky.

V tretej kapitole rozoberám existujúce software-ové riešenia, používané na zefektívnenie práce v advokátskej kancelárii. Zaujímajú ma predovšetkým služby poskytované týmito aplikáciami a ich možné rozšírenia, či úpravy.

V štvrtej kapitole uvádzam jednotlivé časti mojej aplikácie z hľadiska funkcionality a možného využitia.

¹odkaz: <http://www.orsr.sk>

Piata kapitola je venovaná konceptuálnemu návrhu štruktúry systému pomocou nástrojov software-ového inžinierstva. Tento návrh ponúka pevný základ pre následnú implementáciu využitím nástrojov uvedených v kapitole číslo šesť.

Kapitola šesť obsahuje opis implementácie aplikácie.

V poslednej kapitole číslo sedem ponúkam zhrnutie mojej práce, dosiahnuté výsledky a rozoberám možné rozšírenia.

Kapitola 2

Legislatívne a administratívne požiadavky

Pre vývoj vhodnej aplikácie pre advokátsku prax je potrebné zohľadniť niekoľko aspektov.

Jeden zo zohľadňovaných aspektov tvoria legislatívne požiadavky predpísané zákonmi Slovenskej republiky. Tieto legislatívne požiadavky definujú najmä spôsob zabezpečenia osobných údajov a ich spracovanie. Taktiež stanovujú povinnosť prevádzkovateľa informačného systému vykonať určité právne úkony najmä pre účely evidencie daného systému. Bližší opis týchto požiadavok sa nachádza v prvej podkapitole.

Analýzu legislatívnych požiadavok som konzultoval s JUDr. Miloslavom Hrickom.

Druhým z aspektov sú požiadavky vyplývajúce z bežných pracovných úkonov prebiehajúcich počas výkonu advokácie na území Slovenskej republiky. Tieto pracovné úkony a ich implikácie na následný vývoj aplikácie sú preberané v druhej podkapitole.

Analýzu pracovných procesov v advokátskej kancelárii som konzultoval s JUDr. Jánom Bartánusom.

V tretej kapitole ponúkam analýzu súčasného stavu Obchodného registra Slovenskej republiky, z hľadiska jeho zamýšľanej funkcie, štruktúry a legislatívy plynúcej z viacerých oficiálnych zdrojov.

Táto kapitola nám teda môže poskytnúť základnú predstavu o budúcom systéme.

2.1 Legislatívne požiadavky

Bežný advokát sa pri svojej práci skoro denne stretáva s rôznymi formami spracovania osobných údajov klientov. Tieto údaje sú citlivé a ich spracovanie je chránené zákonmi vydanými Úradom na ochranu osobných údajov Slovenskej Republiky.

V nasledujúcich častiach tejto podkapitoly sa venujem povinnostiam pri zavedení informačného systému určeného na spracovávanie takýchto údajov a nutným procedúram, ktoré by mali dostatočne zabezpečiť spracovávané citlivé údaje.

2.1.1 Evidencia informačného systému

Informačné systémy, ktoré spracúvajú osobné údaje podliehajú určitým povinnostiam uvedeným v § 33 a nasledujúcich, Zákona č. 122/2013 Z. z. o ochrane osobných údajov.

Podľa typu informačného systému môže tento podliehať: [1]

- Oznámeniu:

- Všetky informačné systémy, v ktorých sa osobné údaje spracúvajú automatizovane alebo čiastočne automatizovane. Vlastne akýkoľvek systém, čo nie je papierová kartotéka.
- Výnimky z tohto sú:
 - * Ak sa informačný systém musí registrovať.
 - * Ak má prevádzkovateľ zodpovednú osobu, ktorá má príslušnú skúšku. Z tohto je však ďalšia výnimka, a to ak by šlo o kamerový systém.
 - * Ak sa osobné údaje spracúvajú na základe zákona.
 - * Ak ide o informačný systém občianskych združení, cirkví, odborov a obsahuje iba údaje o členoch týchto organizácií.

- Registrácii:

- Kamerový informačný systém, ak o tom rozhodne úrad.
- Spracovanie osobných údajov nie na základe zákona, ale na základe súhlasu dotknutej osoby.
- Bez súhlasu dotknutej osoby, za účelom plnenia zmluvy.

- Evidencii:

- Ak nepodlieha oznámeniu alebo registrácii.
- Prevádzkovateľ systému vyplní evidenčný list a založí si ho.

Na základe týchto informácií som najprv usúdil, že informačný systém vyvíjaný v rámci tejto bakalárskej práce bude nutné oznámiť na úradoch. Po konzultácii s advokátom JUDr. Miloslavom Hrickom, som sa však dozvedel, že podľa §18 odsek 6 a 7, Zákona č.586/2003 Z. z. o advokácii [2] sa udeľuje výnimka pre advokátov.

Advokáti môžu spracovávať osobné údaje na základe spomínaného zákona, a preto informačný systém podlieha iba nutnosti evidencie.

Evidencia sa podľa § 43–§ 44 zo zákona o ochrane osobných údajov musí viesť od počiatku spracúvania týchto údajov v danom informačnom systéme až po jeho ukončenie.[1] Evidenciu vedie prevádzkovateľ informačného systému, čiže zodpovedný advokát.

Evidencia o informačnom systéme musí obsahovať:[1]

- Identifikačné údaje prevádzkovateľa.
- Meno a priezvisko štatutárneho orgánu prevádzkovateľa alebo inej osoby oprávnenej konať v mene prevádzkovateľa.
- Identifikačné údaje zástupcu prevádzkovateľa, ak je vymenovaný; ak prevádzkovateľ vymenoval svojho zástupcu, uvedie aj meno a priezvisko štatutárneho orgánu zástupcu prevádzkovateľa alebo inej osoby oprávnenej konať v mene zástupcu prevádzkovateľa.
- Počet oprávnených osôb prevádzkovateľa.
- Názov informačného systému.

- Účel spracúvania osobných údajov.
- Právny základ spracúvania osobných údajov.
- Okruh dotknutých osôb.
- Zoznam osobných údajov, alebo rozsah osobných údajov.
- Tretie strany, prípadne okruh tretích strán, ak sa predpokladá alebo je zrejmé, že im budú osobné údaje poskytnuté a právny základ ich poskytovania.
- Okruh príjemcov, ak sa predpokladá alebo je zrejmé, že im budú osobné údaje sprístupnené a právny základ ich sprístupnenia.
- Spôsob zverejnenia, ak prevádzkovateľ osobné údaje zverejňuje a právny základ ich zverejnenia.
- Tretie krajiny, ak sa predpokladá alebo je zrejmé, že sa do týchto krajín uskutoční prenos osobných údajov a právny základ ich prenosu.
- Označenie bezpečnostných opatrení prijatých na zabezpečenie ochrany osobných údajov podľa § 19 odsek 1 a 2.
- Dátum začatia spracúvania osobných údajov v informačnom systéme.

Táto časť prieskumu slovenskej legislatívy teda preukázala, že počas doby používania informačného systému vyvíjaného v rámci tejto bakalárskej práce, spadá zodpovednosť za povinnosť vedenia evidencie informačného systému na advokáta, ktorý ho využíva.

2.1.2 Bezpečnostné požiadavky

Napriek tomu, že ako vývojár informačného systému pre advokátske kancelárie som oslobodený od oznamovacej a evidenčnej povinnosti, ako bolo preukázané v časti 2.1.1 tejto podkapitoly, je nutné v spolupráci s advokátom, ktorý systém využíva zaistiť splnenie istých bezpečnostných opatrení predpísaných Úradom o ochrane osobných údajov v prílohe k vyhláške číslo 164/2013 o rozsahu a dokumentácii bezpečnostných opatrení.[3]

Tieto opatrenia sa delia na technické opatrenia a opatrenia organizačné.

Medzi prvý typ opatrení spadá napríklad zabezpečenie informačného systému prostriedkami fyzickej povahy, ako je ochrana pred krádežou, či nepriaznivými vplyvmi prostredia.

Na realizáciu tohto druhu zabezpečenia je najvhodnejším spôsobom umiestnenie nosiča¹ priamo do advokátskej kancelárie, do pripraveného trezoru. Tento spôsob vedenia informačného systému zaisťuje prístup len osobám, ktoré majú prístupový kód, alebo kľúč od trezoru a chráni nosič pred väčšinou predvídateľných vonkajších vplyvov, ktoré by mohli negatívne ovplyvniť uložené údaje.

Technické opatrenia zahŕňajú bezpečnostné požiadavky z hľadiska technickej implementácie. Tieto požiadavky sú napríklad ochrana pred neoprávneným prístupom, či riadenie prístupu oprávnených osôb. Tento problém navrhujem v mojom systéme riešiť enkrypciou disku, alebo súboru, kde sa budú nachádzať citlivé údaje informačného systému a autentizáciou užívateľov, za pomoci hesiel, ktoré budú uložené v lokálnej databáze v zahashovanom tvare pomocou pätnástich iterácií algoritmu bcrypt, založenom na šifre Blowfish. [24]

¹serveru

Ochrana proti škodlivému kódu bude zaistená napríklad pomocou odhadovania prípon súborov nahrávaných do systému, podľa ich obsahu. Mená súborov budú generované systémom. Užívateľ systému bude môcť súbory iba nahrávať a sťahovať.

Zálohovanie dát bude prebiehať pravidelne buď na iné miesto využívaného disku, alebo na náhradný disk, ak je takýto k dispozícii. Táto záloha, spolu s pôvodnými dátami môže byť v prípade potreby likvidácie vymazaná pomocou predpripraveného príkazu.

Informačný systém taktiež musí byť zavedený na stabilnej verzii operačného systému spolu s aktualizovanými nástrojmi, využívanými na vedenie serveru.

Organizačné opatrenia sa zaoberajú najmä školením užívateľov ohľadom správneho užívania informačného systému, za účelom zaistenia zabezpečenia citlivých údajov. Ide napríklad o odporúčenie úrovne náročnosti prístupových údajov pri tvorbe účtu a spôsob narábania s priradenými údajmi. Jedným spôsobom zabezpečenia dostatočnej úrovne hesiel pri zavedení systému je aj náhodná generácia zložitých hesiel pre užívateľov, čo zamedzí možnosti slovníkových útokov na systém.

Systém musí taktiež ponúkať rôzne úrovne prístupu (role) k citlivým údajom, prípadne k možnosti ich ďalšieho spracovania v závislosti od pracovnej pozície v advokátskej kancelárii.

Zároveň je vhodné zaučiť užívateľov ohľadom postupov pri bezpečnostnom incidente, či nutnosti likvidácie osobných údajov. V takomto prípade by bolo najvhodnejšie oznámiť udalosť vedeniu, ktoré by následne kontaktovalo administrátora ohľadom ďalších krokov.

Je taktiež užitočné zaviesť kontrolnú činnosť, a to najmä záznam prístupov do systému. Na tento účel navrhujem do záznamového súboru vytvoreného systémom zapisovať čas prístupu do systému a identitu prístupujúcich zamestnancov.

2.2 Procesy v advokátskej kancelárii

Najväčší vplyv na návrh informačného systému, vyvíjaného v rámci tejto bakalárskej práce, má skutočný priebeh práce zamestnanca advokátskej kancelárie.

Zamestnanec počas výkonu advokácie reaguje na požiadavky klientov, ktorí navštevujú advokátsku kanceláriu hľadajúc právnu pomoc. Advokát prijme a usadí klienta v kancelárii. Klient mu následne vysvetlí problém, ktorý má, prípadne rovno špecifikuje službu, ktorú od advokáta očakáva.

Advokát následne analyzuje situáciu. Najskôr sa snaží zistiť, či je klientovi v jeho možnostiach a schopnostiach pomôcť. Prípady v ktorých klientovi pomôcť nemôže, sú napríklad zameranie na inú oblasť práva, či žiadosť o službu, ktorá nekorešponduje so súčasne platným zákonom.

Ak je advokát schopný klientovi pomôcť, vysvetlí mu situáciu a opíše očakávaný postup výkonu práce pre klienta. To zahŕňa približný počet úkonov, požadované dokumenty a v konečnom dôsledku aj predpokladanú odmenu za jednotlivé úkony, ktorá sa dá vypočítať podľa vyhlášky Ministerstva spravodlivosti Slovenskej republiky 655/2004 Z. z. o odmenách a náhradách advokátov.

Typy odmien môžu byť napríklad hodinová, podielová, či paušálna. Tieto typy odmien závisia od dohody s klientom, ktorý sa môže rozhodnúť pre odmenu, ktorá mu v závislosti od požadovaných služieb vyhovuje najviac. Pokiaľ klientovi ani jeden z troch uvedených typov odmien nevyhovuje, pristupuje sa k odmene tarifnej, ktorá je určená spomínanou vyhláškou podľa typu a počtu vykonaných úkonov počas plnenia služby pre klienta.^[4] Advokát taktiež klienta upozorní, že počet úkonov a výška odmeny sa môžu počas výkonu danej práce meniť.

Ak klient súhlasí s predostretým návrhom riešenia jeho požiadavky a predpokladanou odmenou advokátovi, tak v poslednom kroku podpíše plnú moc. Plnou mocou klient prenecháva advokátovi schopnosť jednať v jeho mene, na právne účely.

Po interakcii s klientom advokát vytvorí pracovný spis. Pracovný spis, v závislosti od postupu jednotlivých advokátskych kancelárií, môže obsahovať tieto údaje:

- Údaje o klientovi.
- Údaje o protistrane. (Ak nejaká v danej požiadavke figuruje.)
- Predmet právneho zastúpenia.
- Spisová značka.
- Dátum založenia.

Počas práce na spise sa vykonávajú rôzne úkony, ako napríklad tvorba žalôb, účasť na pojednávaniach, alebo rokovanie s protistranou. Mnohé z týchto úkonov vyžadujú vypracovanie, či priloženie určitých potrebných dokumentov.

Advokát môže v rámci advokátskej kancelárie spolupracovať s inými advokátmi, prípadne zamestnávať koncipientov, či sekretárky. Na spisoch môžu pracovať spoločne, no aj individuálne.

Napriek tomu, že podľa §3 a §8, Zákona 586/2003 Z.z. o advokácii, nesmie byť advokát zamestnaný iným advokátom,^[2] sa občas z organizačných dôvodov jeden z advokátov v kancelárii považuje za „vedúceho“.

Počas výkonu svojej práce môžu zamestnanci advokátskej kancelárie využívať rôzne jednoduché elektronické nástroje na zjednodušenie, či urýchlenie ich práce, ktorú by inak museli vykonávať zdĺhavo a manuálne.

Príkladmi týchto nástrojov sú:

- Kalkulačka na počítanie tarifnej odmeny za úkony.²
- Kalkulačka na úroky z omeškania.²
- Výpočet cestovných náhrad pri použití vlastného vozidla.³
- Obchodný register na internete.⁴

Práve obchodnému registru a jeho vyhľadávaču sa venujem v nasledujúcej podkapitole.

2.3 Obchodný register

Táto podkapitola čerpá údaje z [16].

Obchodný register je verejný zoznam, do ktorého sa zapisujú zákonom stanovené údaje týkajúce sa podnikateľov, prípadne iných osôb, o ktorých to ustanovuje osobitný zákon.

Právnické osoby majú povinnosť zapísať sa do obchodného registra udelenú zákonmi Slovenskej republiky. Fyzické osoby, ktoré podnikajú, sa do obchodného registra zapisujú buď na vlatnú žiadosť, alebo ak tak určí špecifický zákon.

²odkaz: <http://advok.sk/>

³odkaz: <http://www.ekonomika.sme.sk/kalkulacky/cestovne-nahrady-pri-pouziti-vlastneho-vozidla.php>

⁴odkaz: <http://www.orsr.sk>

Všetky osoby sa zapisujú do registra, spravovaného okresným súdom v sídle krajského súdu, v obvode ktorého buď sídlia, podnikajú, alebo majú bydlisko. To, že obchodný register obsahuje pravidelne obnovované údaje o firmách a fyzických osobách, ho robí nenahraditeľnou pomôckou v oblasti obchodného práva.

Vyhľadávanie v obchodnom registri na internete, ktoré vzniklo na základe zmluvy medzi Ministerstvom spravodlivosti Slovenskej republiky a spoločnosťami Microsoft, Compaq a Wedia[23] poskytuje advokátom venujúcim sa obchodnému právu spôsob, ako jednoducho získať údaje z obchodného registra bez nutnosti navštíviť krajský súd.

Poskytuje viacero druhov vyhľadávania:

- Podľa obchodného mena.
- Podľa identifikačného čísla (IČO)⁵.
- Sídlia.
- Spisovej značky.⁶
- Priezviska a mena osoby.

Počas výkonu svojej práce musí advokát pozorne sledovať zmeny v údajoch rôznych osôb, ktoré sú dôležité pre jeho aktuálne zákazky. Keďže týchto osôb môžu byť desiatky a každý pracovník musí sledovať zmeny vo vlastnej sade osôb, napadlo ma tento proces zautomatizovať. Preto môj informačný systém bude využívať vyhľadávacie služby obchodného registra na internete a v pravidelných intervaloch kontrolovať zmeny v záznamoch, ktoré pracovník advokátskej kancelárie označí v systéme na sledovanie.

2.4 Záver

Vďaka analýze legislatívnych aspektov som sa dozvedel o nutných súčastiach informačného systému a povinnostiach, ktoré musím dodržať, pokiaľ má byť výsledná aplikácia vhodne a bezpečne nasadená do prostredia advokátskej kancelárie.

Konzultácia s JUDr. Jánom Bartánusom mi pomohla pochopiť hierarchiu zamestnancov a pracovné procesy v jeho advokátskej kancelárii. Predstavený vzor takejto kancelárie budem ďalej využívať pre účely návrhu a následnej implementácie vyvíjaného informačného systému.

Oboznámenie sa s obchodným registrom Slovenskej republiky mi pomohlo sa dozvedieť viac o jeho funkcii, obsahu a roli, ktorú hrá v advokácii na Slovensku. Internetová služba, využívaná na vyhľadávanie záznamov v obchodnom registri je jednoduchá na používanie a dostatočne užívateľsky prívetivá. Táto služba však neobsahuje možnosť sledovať zmeny v záznamoch. Túto funkciu plánujem implementovať.

⁵IČO – identifikačné číslo organizácie. IČO je osem miestne číslo využívané na jednoznačnú identifikáciu právnických osôb a podnikateľov.[5]

⁶Spisová značka sa využíva na jednoznačné označenie, pod ktorým je u súdu vedená určitá vec, v tomto prípade záznam v obchodnom registri.[9]

Kapitola 3

Existujúce software-ové riešenia

Advokáti výkon svojej práce realizujú rôznymi spôsobmi. Mnohí z nich aj v súčasnej dobe využívajú mechanické pomôcky, ako napríklad správu spisov v papierových kartotékach. Niektorí na spravovanie pracovných súborov využívajú štandardné adresárové štruktúry poskytované operačným systémom, ktorý používajú. Advokáti, ktorých zaujímajú moderné riešenia, majú možnosť zaplatiť za špecializovaný advokátsky software.

Tieto špecializované software-ové riešenia môžu poskytovať širokú škálu užitočných funkcií a nástrojov, ktorými si advokát môže uľahčiť prácu. Niektoré z nich však obsahujú aj rozšírenia, týkajúce sa iných oborov ako je advokácia. Iné riešenia môžu zas pre advokáta byť z rôznych dôvodov nepraktické, pokiaľ ich implementácia prináša určité obmedzenia, či nedostatky.

Niekoľko užitočných aplikácií bolo vytvorených aj pre účely advokácie na Slovensku. V tejto kapitole ich analyzujem a bližšie skúmam ich výhody, aj nevýhody. Informácie o týchto aplikáciách som čerpal z propagačných stránok týchto aplikácií, či priamo webových stránok, kde ich po registrácii môže advokát okamžite využívať.

3.1 LEGUS

The screenshot displays the LEGUS interface for a case titled "Spis - OSP" with ID "090000509". It includes buttons for "ZMENIŤ KONANIE", "ZMAZAŤ KONANIE", and "STIAHNŮŤ CELÝ SPIS". A summary table shows financial and procedural details. Below is a "Chronológia úkonov" (Timeline of actions) table and a "Vyberte si prosím nasledujúci úkon:" (Select the next action) menu.

Isťina (hrubá)	Úrok	Deň splatnosti	V omeškani	Súd. popl. (uhr.)	Aktuálna Isť. (uhr.)
1000 € (1 072,00 €)	18,00 %	02.01.2009	146 dní	100,00 € (50,00 €)	1 072,00 € (0,00 €)

Strany	v zastúpení:
Zalobca TEST s.r.o.	Test JUDr. Huran Robert
Zalovaný Test Pučík Jožko	Test Z Peter

Chronológia úkonov :

Číslo	Úkon	Dátum	Stav
1.	Zadanie	2009-05-28 12:02:40	
2.	Výzva na zaplatenie (pokus o zmier)	2009-05-28 12:02:42	ostáva 7 dní DORUČENÉ
3.	Zaevidovanie šiadosti o splátkový kalendár	2009-05-28 12:16:24	
4.	Schválenie splátkového kalendára	2009-05-28 12:16:25	
5.	Návrh na vydanie platobného rozkazu	2009-05-28 12:23:00	
6.	Čiastočné pinenie žalovaného	2009-05-28 12:27:08	
7.	Úplné pinenie žalovaného	2009-05-28 13:36:43	
8.	Úhrada súdneho poplatku	2009-05-28 20:06:19	
9.	Predvolanie na pojednávanie	2009-05-28 20:18:11	
10.	Zastupovanie na pojednávaní	2009-05-28 20:18:13	
11.	Vyčistenie trov konania	2009-05-28 20:18:15	
12.	Rozsudok okresného súdu	2009-05-28 21:27:28	
13.	Sledovanie vykonateľnosti	2009-05-28 21:27:34	SLEDUJEM (ešte 15 dní)

Vyberte si prosím nasledujúci úkon:

- Výzva na zaplatenie (pokus o zmier) (4)
- Schválenie splátkového kalendára (5)
- Vyjadrenie žalovaného (12)
- Zastupovanie na pojednávaní (28)
- Sledovanie vykonateľnosti (29)

Obr. 3.1: Zobrazenie rozhrania správy spisov v systéme LEGUS.

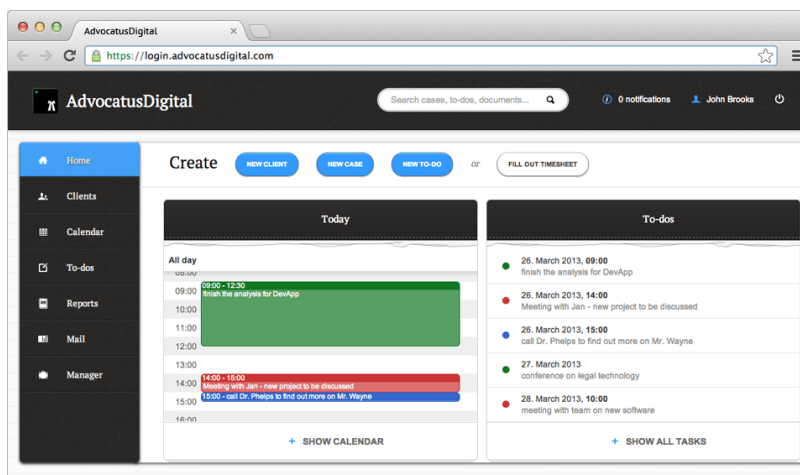
Právny systém LEGUS poskytuje okrem software-u pre advokátske kancelárie aj mnoho ďalšej funkcionality zasahujúcej do iných vládnych pracovných odvetví. Je implementovaný ako modulárna aplikácia s webovým užívateľským rozhraním. Ponúka možnosť zloženia si vlastného systému na mieru, vďaka vstavanej modularite.

Pre advokátov a advokátske kancelárie poskytuje prehľadnú správu spisov, hromadné operácie so spismi, rôzne právne kalkulačky, možnosť vytvorenia vlastných predlôh pre zadávané dokumenty, upozorňovanie na blížiace sa lehoty, či dokonca automatickú fakturáciu klientov.

Správcovia tohto systému ponúkajú bezplatné demo na odskúšanie. K dnešnému dňu systém LEGUS používa 223 aktívnych používateľov.[14]

Obrázok 3.1 som získal z ukážok systému umiestnených na stránke aplikácie.

3.2 AdvocatusDigital



Obr. 3.2: Zobrazenie užívateľského rozhrania systému AdvocatusDigital.

Tento nástroj vyvinutý spoločnosťou DevApp s.r.o. poskytuje jednoduché užívateľské rozhranie na správu informácií týkajúcich sa právnej praxe.

Na webovej stránke tohto software-u je uvedené, že bezpečnosť údajov je implementovaná rovnakým spôsobom, ako bezpečnosť údajov v službe internet bankingu. Tieto údaje je možné sprístupniť klientovi.

Kalendár, ktorý je možné viesť pomocou aplikácie AdvocatusDigital je možné prepojiť s vlastným kalendárom Google. V rámci systému je možné prijímať a odosielať poštu a udržiavať si prehľad o čase strávenom na jednotlivých úlohách.

AdvocatusDigital ponúka centralizované rozhranie na vyhľadávanie vlastných klientov, prípadov, úloh, alebo záznamov v rámci aplikácie. Rovnako ako systém LEGUS je možné si túto aplikáciu bezplatne vyskúšať. Jej implementácia sa však zdá byť zameraná viac medzinárodne, než priamo na účely slovenskej advokátskej kancelárie.[6]

Obrázok 3.2 užívateľského rozhrania som získal z ukážky systému, nachádzajúcej sa na titulnej stránke software-u AdvocatusDigital.

3.3 Tribunal

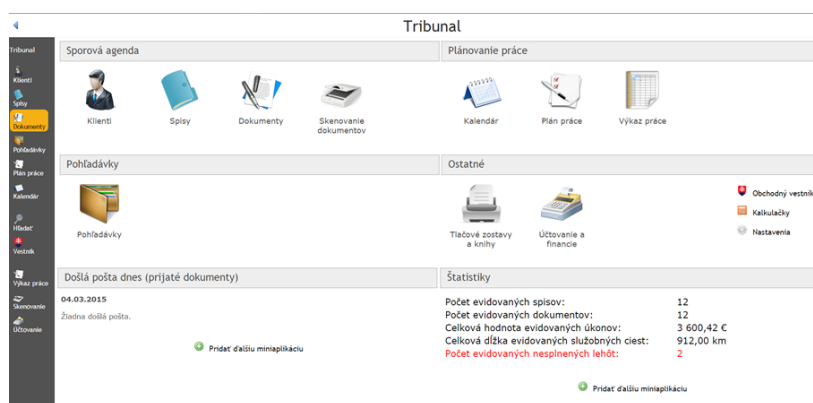
Software Tribunal, implementovaný ako desktopová aplikácia sa na vlastnej stránke prezentuje ako komplexné riešenie, pokrývajúce administratívu advokátskej kancelárie. Je v ňom možné prehľadným spôsobom viesť agendu klientov, spisov, dokumentov a úloh.

Taktiež umožňuje vedenie si vlastného kalendáru udalostí, ktoré sa následne zapisujú do podrobného žurnálu. Rovnako ako systém LEGUS, ponúka možnosť vyúčtovania právnych úkonov a následnú fakturáciu klienta. Na tento účel taktiež ponúka aktuálne kalkulačky na výpočet trovov právneho zastúpenia, úrokov z omeškania, či súdnych poplatkov.

Tribunal poskytuje možnosť 30-dňovej skúšobnej verzie. Systém je prístupný na základe nastavených oprávnení a jednotliví užívatelia sa v ňom môžu navzájom zastupovať. Tribunal umožňuje výber medzi hostingom v cloude, virtuálnym serverom a inštaláciou priamo v kancelárii.

Výhodnou službou, ktorú Tribunal ponúka, je možnosť školenia zamestnancov zdarma.[21]

Obrázok 3.3 užívateľského rozhrania som získal z ukážok systému umiestnených na stránke aplikácie.



Obr. 3.3: Zobrazenie úvodnej obrazovky software-u Tribunal.

3.4 ADVOS

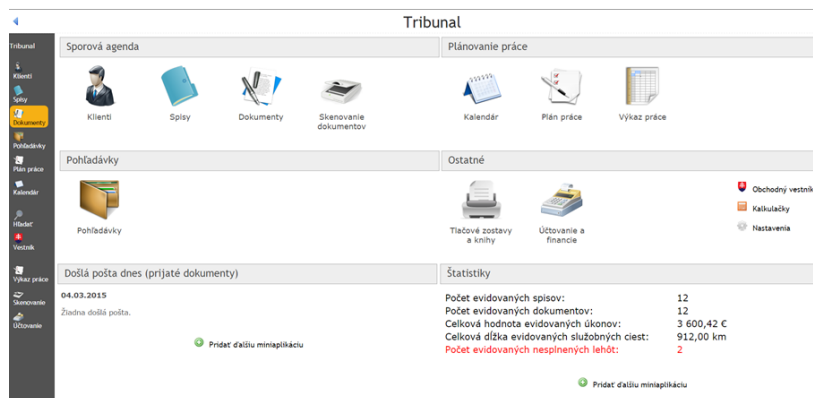
Informačný systém ADVOS, určený pre prácu v advokátskej kancelárii je, podobne ako systém Tribunal, implementovaný formou desktopovej aplikácie. Podobne ako ostatné systémy ponúka správu osôb, spisov, dokumentov a nadchádzajúcich udalostí.

Ponúka nástroje na výpočet odmeny advokáta za právne úkony a umožňuje integrácie dát z užívateľom zvoleného e-mailového klienta.

Výhodou tohto systému je vedenie žurnálu práce advokátskych koncipientov a jej výpis pre potreby advokátskej komory. Vďaka tomu sa advokát môže plne venovať svojej práci, namiesto zložitej administratívy práce koncipienta.

Rovnako ako systém Tribunal, poskytuje užívateľské školenia hromadnou, ale aj individuálnou formou. Školenia pre aplikáciu ADVOS sú však platené. Propagačná stránka sa zdá byť od roku 2015 neobnovovaná.[7]

Obrázok 3.4 užívateľského rozhrania som získal z ukážok systému umiestnených na stránke aplikácie.



Obr. 3.4: Zobrazenie správy spisov desktopovej aplikácie ADVOS.

3.5 Zhrnutie

Analýza existujúcich software-ových riešení mi predstavila súčasnú ponuku informačných systémov určených na správu údajov v advokátskej praxi.

V tejto kapitole boli popísané štyri existujúce software-ové riešenia s rôznym zameraním a implementáciou. Každé z nich poskytuje užívateľovi rôznu funkcionálnosť, s rôznym stupňom zamerania na advokátsku prax.

Zhrnutie analyzovaných nástrojov môžete nájsť v tabuľke 3.1.

Nástroj	Implementácia rozhrania	Údaje z verejných databáz	Školenia	Výpočet fakturácie klienta	Spôsob platby
LEGUS	Web	Obchodný vestník	Nie	Áno	Podľa modulov
AdvocatusDigital	Web	Nie	Nie	Nie	Paušálne
Tribunal	Desktop	Obchodný vestník	Zdarma	Áno	Paušálne
ADVOS	Desktop	Obchodný vestník	Platené	Áno	Jednorázovo + Paušálne

Tabuľka 3.1: Zhrnutie vlastností nástrojov.

Kapitola 4

Analýza požiadavok na navrhovaný systém

Súčasťou tejto bakalárskej práce je aplikácia, ktorá pri zavedení do praxe musí zaistiť istý štandard bezpečnosti spracovávaní osobných údajov opísaný v kapitole 2.1.2. Zároveň bude obsahovať vhodne vybrané súčasti už existujúcich riešení uvedených v kapitole 7, ktoré vyplývajú z bežných potrieb štandardnej advokátskej kancelárie. Napojenie na verejné databáze som sa rozhodol voči už existujúcim riešeniam pozmeniť.

Návrh som priebežne konzultoval s pánmi JUDr. Jánom Bartánusom a JUDr. Miloslavom Hrickom.

4.1 Správa spisov advokátskej kancelárie

Správa spisov v advokátskej kancelárii je náročná záležitosť. Bežne sa stretujeme s dvomi spôsobmi spravovania spisov. Prvým z nich je správa spisov manuálne, v papierovej podobe. V prípade papierového spracovania spisov ukladajú advokáti spisy spolu s dokumentami vypracovanými v rámci úkonov do fasciklov. Toto riešenie je však časovo, aj priestorovo náročné.

Pokiaľ advokáti pristúpia k druhému spôsobu, a to organizácii spisov elektronicke, vystavia sa najmä časovému riziku zaškolenia zamestnancov a otázke spôsobu organizácie spisov. Pokiaľ nie je advokát aspoň čiastočne skúsený v práci s počítačom, môže organizáciu adresárovej štruktúry podceniť a následne dennodenne tráviť veľa času hľadaním potrebného dokumentu.

Tento problém som sa rozhodol v mojej aplikácii riešiť ukladaním spisov a ich zložiek do databázy. Užívateľ si bude môcť prehliadať spisy priamo v systéme, spolu s úkonmi potrebnými k dokončeniu spisu. Tieto úkony môžu vyžadovať doloženie určitých dokumentov. Vďaka môjmu systému však užívateľ nebude musieť pri hľadaní dokumentu navigovať cez zložitú adresárovú štruktúru, no pristúpi k nim priamo cez prívetivé užívateľské rozhranie. Tieto dokumenty bude možné v systéme vyhľadávať.

Pre tento prístup som sa rozhodol z dvoch dôvodov. Jedným z nich je abstrakcia adresárovej štruktúry do jednoduchej navigácie v rámci webového rozhrania, čo umožní priamočiary prístup k informáciám, ktoré užívateľ práve potrebuje. Druhým dôvodom je možnosť organizácie práce. Podľa môjho návrhu bude mať užívateľ prístup len k tým spisom, na ktorých pracuje. Pokiaľ potrebuje pristúpiť k dokumentu, ktorý nie je priradený ku spisu,

na ktorom momentálne pracuje, môže sa k nemu dostať pomocou spomínaného vyhľadávania.

Spis musí obsahovať určité údaje rozoberané v kapitole 2.2. Nevyhnutnou súčasťou spisu sú detaily klienta, ktorý žiadal o službu advokáta. Dobrovoľnou súčasťou je protistrana. Oba tieto údaje reprezentujú určitú osobu, buď fyzickú, alebo právnicku. Táto môže byť súčasťou viacerých spisov, jej uloženie do databázy teda umožní jej znovupoužitie v prípade potreby. Ďalšími súčasťami sú spisová značka a predmet právneho zastúpenia. Tieto dva údaje, spolu s akýmkoľvek poznámkami, bude možné uložiť do popisu spisu.

Dokumenty, osoby, úkony a spisy bude možné vytvárať pomocou jednoduchej webovej formy. Každú z uvedených položiek, okrem spisu bude možné zo systému odstrániť. Keď bude spis uzavretý, bude ho možné deaktivovať. Počas toho, ako je spis deaktivovaný v ňom nebude možné nič upravovať, či už priradených zamestnancov, úkony, alebo dokumenty. Takýto spis sa chápe ako archivovaný. V prípade potreby, (napr. výskyt nečakaného úkonu v rámci spisu) bude možné spis znova aktivovať.

Pokiaľ bude zamestnanec priradený ku spisu, alebo ku spisu, ku ktorému je zamestnanec priradený, pribudne nová udalosť, zamestnanec o tom bude upozornený.

4.2 Napojenie na obchodný register

Vyvíjaný informačný systém pre advokátske kancelárie bude schopný vyhľadávať vo verejnej databáze obchodného registra na internete, ako bolo opísané v kapitole 2.3. Obchodný register na internete poskytuje rozhranie na vyhľadávanie záznamov. Je možné vyhľadávať podľa rôznych kritérií.

Obyčajná reprodukcia tohto vyhľadávania by podľa môjho názoru nebola dostatočne prínosná. Preto v mojom systéme túto funkcionálnosť rozširujem o možnosť sledovania záznamov v obchodnom registri. Užívateľ si bude schopný uložiť záznam do databázy na sledovanie. Systém bude potom pomocou pravidelnej kontroly pozorovať zmeny v záznamoch. Pokiaľ takáto zmena nastala, užívateľ bude na to v rámci systému upozornený.

4.3 Pripomienky o udalostiach

Advokáti sa počas svojej práce stretávajú s rôznymi lehotami, či termínmi udalostí. Zvyčajne stolné papierové kalendáre postačujú. Je však praktické mať udalosti generované automaticky podľa zadávaných lehôt ku úkonom priamo v systéme, v ktorom pracujú.

Práve preto som sa rozhodol do aplikácie zahrnúť kalendár udalostí, ktoré užívateľa čakajú. V tomto kalendári sa budú nachádzať automaticky generované udalosti a udalosti ručne vytvorené užívateľom.

4.4 Správa zamestnancov advokátskej kancelárie

Jedna z otázok, ktorou som sa pri návrhu zaoberal, je hierarchia zamestnancov v advokátskej kancelárii. Hierarchická štruktúra sa medzi kanceláriami môže diametrálne líšiť, či sa už jedná o počet zamestnancov, ich vzájomné postavenie, alebo právomoci. V niektorých z kancelárií sú si všetci advokáti rovní a iba spolupracujú. V iných sú niektorí z nich neoficiálne nadradení. Práve z druhého prípadu som z praktických dôvodov v konečnom dôsledku vychádzal a rozhodol som sa, že možnosť správy zamestnancov bude mať v rukách pomyselný vedúci kancelárie.

Vedúci kancelárie bude môcť prijímať aj vyhadzovať zamestnancov. Vyhodenie zamestnanca nemusí byť trvalé, a preto je nutné mať možnosť prijať zamestnanca späť. Zamestnanci budú teda uložení v databáze spolu s ich údajmi.

Okrem vedúceho kancelárie bude systém obsahovať advokátov, ktorí budú mať možnosť zakladať spisy, či priradzovať úkony k spisom. Taktiež budú môcť vytvárať osoby, ktoré budú následne využiteľné ako klient, alebo protistrana.

Zvyšok funkcionality bude dostupný všetkým zamestnancom, a to koncipientom a sekretárkam. Rozhodol som sa tak preto, lebo aj napriek tomu, že sú oficiálne tieto dva typy zamestnancov na dvoch rozličných pozíciách, tak bude vo výkone ich práce zvyšok systému užitočný rovnako.

4.5 Nástroje

Jedna z najjednoduchších, no najužitočnejších častí, môjho systému sú jeho nástroje. Príklady nástrojov som uviedol v kapitole 2.3. Ich zahrnutie do systému poskytne užívateľom ideálne pracovné prostredie pre advokátsku prax, bez nutnosti vyhľadávať niekoľko rozdielnych stránok kvôli elementárnym výpočtom.

4.6 Bezpečnosť

Zavedenie informačného systému pre advokátske kancelárie vyžaduje istú úroveň bezpečnosti opísanú v kapitole 2.1.2. Je nutné realizovať autentifikáciu užívateľov pri vstupe do systému. Tá bude realizovaná jednoduchou prihlasovacou stránkou, ku ktorej ako jedinej, bude mať prístup neprihlásený užívateľ. Po zadaní správneho prihlasovacieho mena a hesla bude užívateľ autentifikovaný a bude mu umožnené vstúpiť do systému.

Jednou z potrebných súčastí je taktiež záloha údajov. Tá môže prebiehať pravidelne podľa periódy určenej advokátom, ktorý chce systém používať. Záloha dokumentov bude z dôvodu šetrenia miestom prebiehať len pokiaľ sa niektorý z dokumentov zmenil, alebo nejaký pribudol.

Záloha údajov bude uložená na špecifikovanom mieste podľa dátumu vytvorenia. Verzie záloh, ktoré budú staršie než rozsah intervalu, ktorý určí advokát, ktorý o zavedenie systému požiadal, budú navždy odstránené.

4.7 Záver

V tejto kapitole som zhrnul moje rozhodnutia určujúce budúci vývoj informačného systému pre advokátsku kanceláriu na území Slovenskej republiky, plynúce z analýzy požiadavok na taký systém. Plánujem vytvoriť aplikáciu sústreďujúcu sa na uľahčenie časovo, či priestorovo náročných úkonov, s ktorými sa zamestnanci advokátskej kancelárie stretávajú v praxi. Chcel by som, aby bola výsledná aplikácia užívateľsky prívetivá a adekvátne na potreby advokátskych kancelárií Slovenskej Republiky.

Kapitola 5

Návrh systému

Analýza požiadavok z kapitoly 4 poskytuje neformálny prehľad o súčastiach vyvíjaného informačného systému. Pred implementáciou je vhodné previesť neformálne požiadavky do vizuálnej podoby, ktorá jasnejšie určí, akým spôsobom bude prebiehať implementácia software-u.

V oblasti software-ového inžinierstva sa na tieto účely zvyčajne používa univerzálny jazyk pre vizuálne modelovanie systémov, UML¹[22].

V nasledujúcich podkapitolách opisujem vizualizáciu systému, realizovanú za pomoci jazyka UML. Každá z nasledujúcich podkapitol obsahuje jeden typ diagramu definovaného v jazyku UML.

5.1 Diagram prípadov použitia

Diagram prípadov použitia sa pri vývoji software-u využíva na popis chovania systému z hľadiska užívateľa. Poskytuje možnosť grafického zobrazenia použitia systému z hľadiska používateľa.[10]

Advokátska kancelária rozlišuje zamestnancov podľa určitej hierarchie, opísanej v predchádzajúcich kapitolách. Táto hierarchia je premietnutá do vytvoreného diagramu prípadov použitia.(viď obrázok 5.1)

Prípady použitia v závislosti od typu aktéra sa dajú zhrnúť takto:

- **Neprihlásený užívateľ** – Neprihlásený užívateľ so systémom nie je schopný interagovať, okrem možnosti sa do systému prihlásiť.
- **Sekretárka/Koncipient** – Tento aktér zahrňuje oba typy zamestnancov obsiahnutých v názve. Títo zamestnanci môžu vykonávať niekoľko rozličných druhov činností:
 - Odhlásenie zo systému.
 - Prehliadanie upozornení.
 - Prehliadanie dokumentov – Počas prehliadania dokumentov môže užívateľ dokumenty vyhľadávať, prípadne s nimi manipulovať. Manipulácia s dokumentami znamená pridávanie nových dokumentov, prípadne mazanie, či úprava už nahradených dokumentov.

¹UML – Unified Modeling Language

- Prehliadanie nástrojov – Pri prehliadaní nástrojov dokáže užívateľ spočítať úroky z omeškania, trovy právneho zastúpenia, či označiť záznam v obchodnom registri ako sledovaný.
 - Prehliadanie priradených spisov – Pokiaľ sa užívateľ rozhodne zobrazíť jemu priradené spisy, môže označiť spis ako uzavretý, pokiaľ sa práca na spise skončila. V opačnom prípade má možnosť zobrazíť jeden z úkonov a tým sa informovať o jeho náplni, prípadne potrebných dokumentoch k dokončeniu úkonu. Ak má takýto dokument k dispozícii, môže ho k úkonu priradiť. Ak sa úkon dokončí, užívateľ ho tak môže označiť.
 - Prezrieť kalendár lehôt – V kalendári lehôt môže užívateľ prehliadať nadchádzajúce lehoty, prípadne vytvoríť vlastnú udalosť.
 - Prehliadať svoj profil – Po zobrazení svojho profilu si ho môže zamestnanec upraviť.
- **Advokát** – Advokát dedí akcie Sekretárky/Koncipienta a rozširuje možnosti interakcie so systémom o tieto akcie:
 - Pridať osobu – Advokát môže pridať do systému buď fyzickú, alebo právnicku osobu
 - Založiť spis – Po založení spisu musí advokát pridať k spisu osobu a úkon. Po pridaní úkonu je potrebné, aby úkon obsahoval poznámku, ktorá bližšie špecifikuje, čo je nutné vykonať. Taktiež je možné špecifikovať, aké dokumenty sú k dokončeniu úkonu potrebné. Nakoniec môže Advokát určiť zamestnancov, ktorí na spise majú pracovať.
 - **Vedúci kancelárie** – Vedúci kancelárie môže vykonávať rovnaké činnosti ako advokát, no má moc nad správou zamestnancov. Môže zamestnancov prijímať, ale aj prepúšťať.

5.2 Entitno-relačný diagram

Diagramy vytvorené pomocou metódy entitno-relačného modelovania dajú sa nazývať entitno-relačné diagramy (ERD, ER) [13].

Môj systém obsahuje množinu dôležitých entít. Jednou z najdôležitejších entít v databáze je Zamestnanec. Dáta o zamestnancoch musia byť v databáze uchovávané z dôvodu organizácie zamestnancov, ich identifikácie v rámci systému a možnosti vytvorenia vzťahu s entitami, ktoré držia údaje relevantné ku konkrétnemu zamestnancovi. Pozíciu zamestnanca určuje jeho Typ zamestnanca. Zamestnanec môže byť upozornený na zmenu v Záznamoch, ktoré sleduje. Údaje o Zázname určujú, aký záznam je v obchodnom registri sledovaný. Zamestnanec môže byť taktiež priradený ku Spisom, na ktorých pracuje. Spis musí obsahovať klienta a môže obsahovať protistranu. Klient aj protistrana sú vyjadrení entitou Osoba, ktorá môže byť fyzická, alebo právnická. Spis pozostáva z niekoľkých úkonov. Úkon musí mať určený typ. Počas práce na úkone môže byť potrebné priradiť pracovné dokumenty. Takúto požiadavku o priradenie dokumentu vyjadruje entita Žiadaný dokument. Žiadosti o dokument môže byť vyhovieť, a to priradením potrebného dokumentu. Ku žiadosti o dokument môže byť priradený len dokument rovnakého typu, určený Typom dokumentu.

V momente priradenia zamestnanca ku spisu, či priradenia žiadosti o dokument, alebo úkonu k spisu, je zamestnanec na túto udalosť upozornený.

Znázornenie tohto modelu formou entitno-relačného modelu je možné vidieť na obrázku 5.2.

5.3 Diagram interakcie so systémom

Na znázornenie možnej užívateľskej interakcie so systémom je vhodné využiť sekvenčný diagram. Sekvenčný diagram je typ diagramu jazyka UML, ktorý zobrazuje chovanie systému usporiadané v čase.[17]

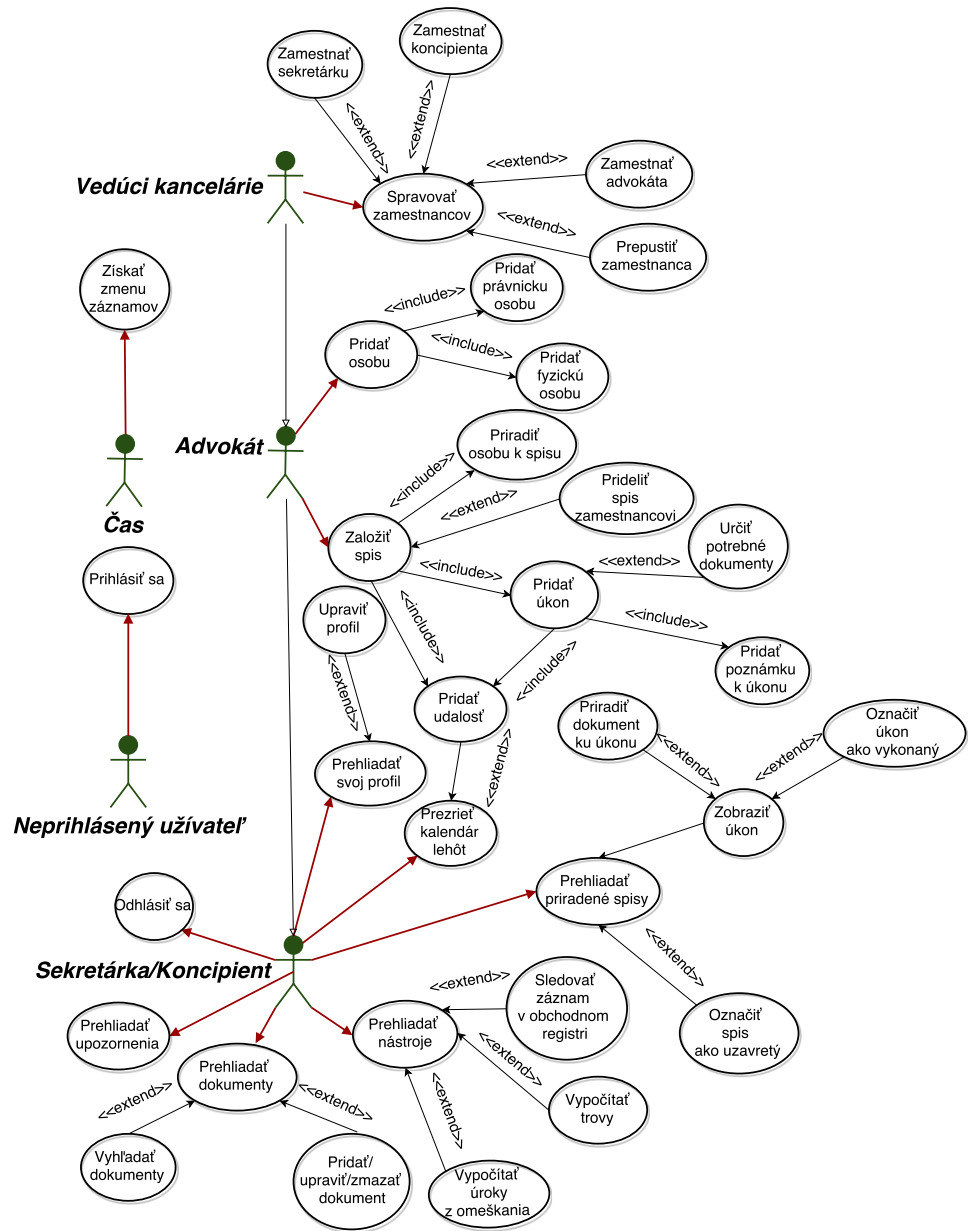
Diagramy 5.3 a 5.4 znázorňujú príklady prácu užívateľa so systémom. V diagrame 5.3 je predstavený spôsob, akým vedúci môže spravovať zamestnancov v systéme. V nasledujúcom diagrame 5.4 je zobrazený možný scenár interakcie zamestnanca s navrhovanou aplikáciou.

5.4 Záver

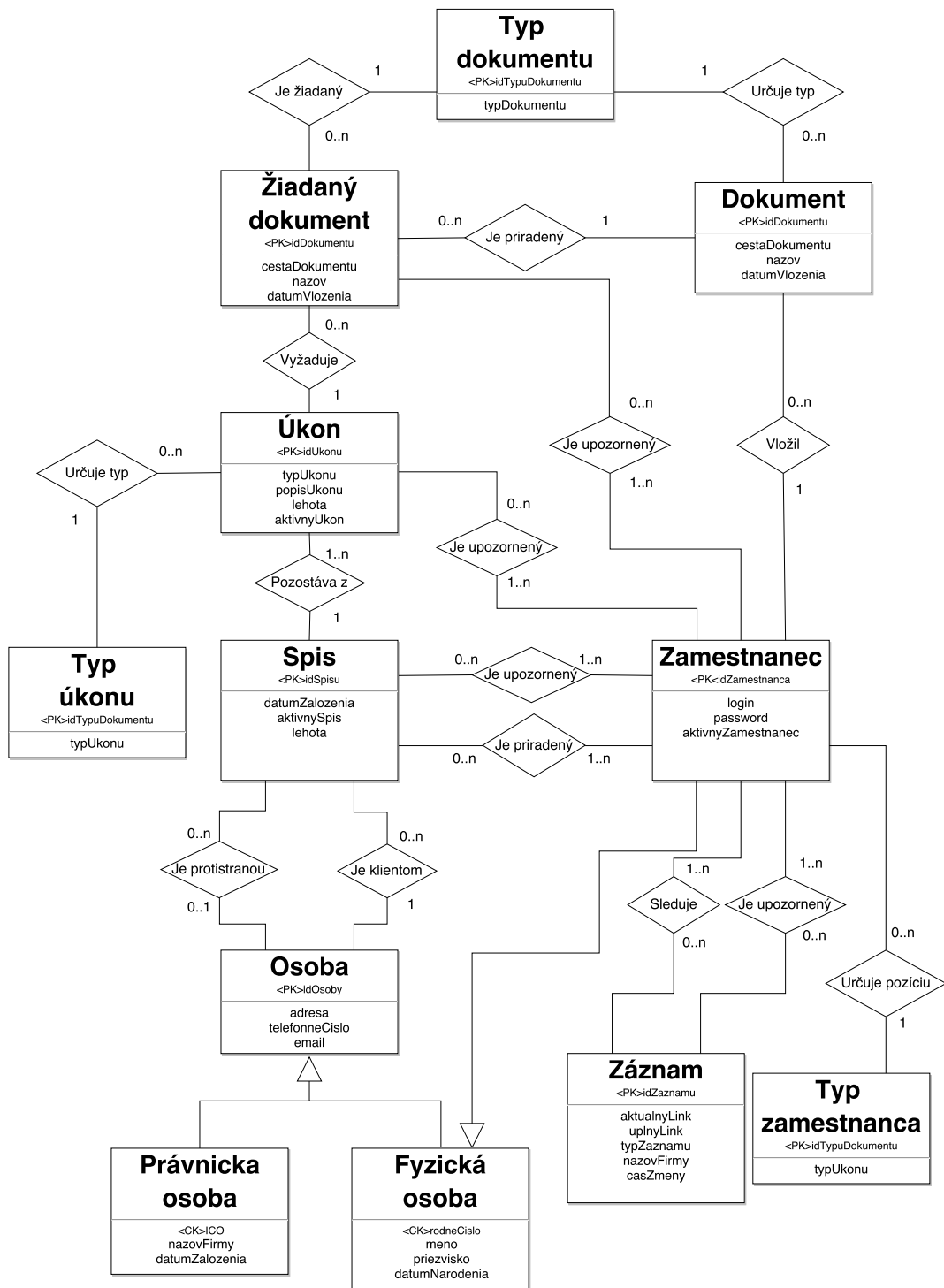
Využívanie modelovacích metód software-ového inžinierstva značne uľahčuje následnú implementáciu systému. Vizualizácia konceptuálnych modelov pomocou jazyka UML umožňuje špecifikovať očakávané správanie a vlastnosti systému.

Diagramy boli vytvorené pomocou nástroja draw.io².

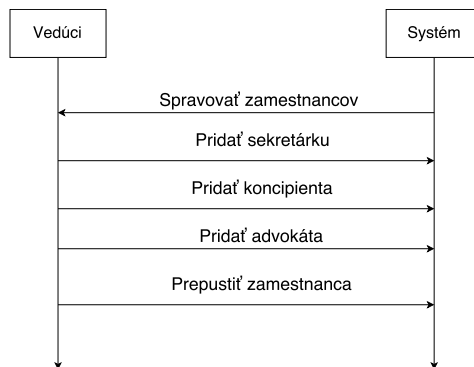
²odkaz: <http://www.draw.io>



Obr. 5.1: Diagram prípadov použitia.



Obr. 5.2: Entito relačný diagram.



Obr. 5.3: Sekvenčný diagram príkladu interakcie vedúceho kancelárie so systémom.



Obr. 5.4: Sekvenčný diagram príkladu interakcie zamestnanca so systémom.

Kapitola 6

Implementácia systému

V tejto kapitole sa venujem spôsobu implementácie konečnej aplikácie.

Na vývoj webovej aplikácie som využíval Open source PHP framework Symfony.[20] Čiastočne som nasledoval MVC¹ návrhový vzor. Najlepším zdrojom informácií počas programovania informačného systému boli online dokumentácie projektov Symfony, Doctrine a Twig.

6.1 Štruktúra programu

Využívanie frameworku Symfony ponúka možnosť vyvíjania aplikácie, ktorá je rozšíriteľná a štruktúrovaná.[19]

Pri zachovaní súborovej štruktúry, odporúčanej najlepšimi praktikami Symfony, je možné zaistiť následnú rozšíriteľnosť aplikácie, prípadne jej čiastočnú modulárnosť. Preto som sa snažil tieto praktiky dodržať a vyvinúť systém, ktorý je kompatibilný so štandardnou štruktúrou aplikácii vyvíjaných v Symfony.

Koreňový adresár mojej aplikácie má názov `/law_office/`. Všetky adresáre(okrem koreňového) spomínané v tejto podkapitole sa nachádzajú v tomto adresári. Štruktúra súborov nachádzajúcich sa v koreňovom adresári, v ktorých sa nachádzajú mnou implementované, či upravované časti aplikácie vyzerá nasledovne:

- `/app/` – Tento adresár obsahuje dva dôležité podadresáre:
 - `/config/` – V tomto podadresári môžeme nájsť konfiguračné súbory vo formáte YAML². Ich hlbší opis sa nachádza v podkapitole 6.2.
 - `/Resources/views/` – V tomto podadresári sa nachádzajú views. Views v MVC chápaní sú vizuálne reprezentácie výstupných údajov, spolu s užívateľským rozhraním.[15] V mojej aplikácii sú implementované ako HTML šablóny vyvíjané pomocou frameworku na šablóny Twig.
- `/src/AppBundle` – adresár `/src/` obsahuje jadro mojej aplikácie.
 - `/Command/` – Tu sa nachádzajú mnou naprogramované príkazy, ktoré je možné volať z príkazového riadku.

¹MVC – Model-view-controller

²YAML – YAML Ain't Markup Language

- * `/BackupDeleteCommand.php` – Tento príkaz vykoná odstránenie zálohy s pred počtu dní definovanom v konfiguračnom súbore.
 - * `/CheckRecordUpdatesCommand.php` – Zadaním príkazu aplikácia skontroluje zmenu sledovaných záznamov v obchodnom registri a následne zmení a pridá relevantné údaje do databázy.
 - * `/CreateBackupCommand.php` – Zadaním príkazu aplikácia vytvorí zálohu dokumentov a databázy.
 - * `/InitSystemCommand.php` – Po zadaní príkazu aplikácia inicializuje prostredie pre beh systému. Vytvorí takmer prázdnu databázu s predvyplnenými údajmi typov zamestnancov, úkonov a dokumentov. Taktiež do systému pridá vedúceho kancelárie s prihlasovacím menom a heslom, ktoré sú zadané ako argumenty pri vyvolaní príkazu.
 - * `/SystemRestoreCommand.php` – Príkaz obnoví dokumenty a databázu zo zálohy uloženej pred ľubovoľným počtom dní. Počet dní je jediným parametrom tohto príkazu.
- `/Controller/` – V tejto časti sa nachádzajú controllers. Controllers sú metódy triedy rozširujúcej triedu Controller, ktoré vo frameworku Symfony prijímajú HTTP požiadavky a zasielajú HTTP odpovede. Medzi týmito úkonmi sa môže nachádzať logika controller-u. Príkladmi tejto logiky sú načítanie, spracovávanie a ukladanie údajov v databáze.[19]
- * `/DocumentController.php` – Operácie s dokumentami.
 - * `/EmploymentController.php` – Správa zamestnancov.
 - * `/EventCalendarController.php` – Funkcionalita kalendára lehôt.
 - * `/LoginController.php` – Autentifikácia užívateľa.
 - * `/LogoutController.php` – Odhlásenie užívateľa.
 - * `/NotificationController.php` – Spracovanie užívateľských upozornení.
 - * `/PersonController.php` – Správa osôb.
 - * `/RecordController.php` – Vyhľadávanie a spravovanie záznamov
 - * `/WorkController.php` – Správa spisov, úkonov, žiadaných dokumentov.
- `/Entity/` – Entities sú triedy reprezentujúce tabuľky v databáze.[20] Na ich vygenerovanie som využil nástroje Doctrine ORM³. Opisu databázového systému sa budem venovať v podkapitole 6.4.
- `/Form/` – V tomto adresári uchovávam triedy generujúce formuláre, pomocou ktorých užívateľ zasiela serveru údaje v štruktúrovanej podobe. Formuláre sú vybudované pomocou objektu typu `FormBuilderInterface`.
- `/Services/` – Services v Symfony sú triedy, ktorých inštancie sa nachádzajú v service containeri. Inštancie services je možné využívať v controller-och a tým umožniť opakovanú využiteľnosť kódu v rámci celej aplikácie.[19] Tento adresár obsahuje mnou naprogramované services.
- * `/databaseFiller.php` – Naplnenie databázy.
 - * `/documentHandler.php` – Spracovanie dokumentov.
 - * `/eventHandler.php` – Vytváranie a získavanie lehôt.
 - * `/notifyHandler.php` – Vytváranie a získavanie lehôt.

³ORM – Object Relational Mapping

- * `/orsrFinder.php` – Vyhľadávač v obchodnom registri.
 - * `/personQuery.php` – Získavanie osôb z databázy.
 - * `/recordQuery.php` – Získavanie údajov z databázy, ktoré sú viazané na záznamy v obchodnom registri na internete.
 - * `/workQuery.php` – Získavanie údajov z databázy, ktoré sa týkajú Spisov.
- `/tests/AppBundle/Services/` – tu sa nachádzajú jednotkové testy mnou implementovaných services.

Externé knižnice, ktoré som využíval počas implementácie aplikácie sa nachádzajú v adresári `/vendor/`. Externé knižnice potrebné pre správnu funkcionálnosť systému, najmä grafického rozhrania, sa nachádzajú v adresári `/web/assets/vendor/`.

6.2 Konfigurácia

V konfiguračnom adresári spomínanom v predchádzajúcej kapitole sa nachádza niekoľko súborov, ktoré je nutné počas zavádzania systému upraviť podľa užívateľských požiadavok.

Súbory a údaje, ktoré je potrebné upraviť sú nasledujúce:

- `/parameters.yml` – Súbor obsahujúci parametre prístupu do databázy.
 - `database_host` – Parameter označujúci URL databázy.
 - `database_port` – Port databázy.
 - `database_name` – Meno databázy.
 - `database_user` – Užívateľ databázy.
 - `database_password` – Heslo od databázy.
- `/config.yml` – Tento súbor obsahuje rôzne konfiguračné parametre vývojového prostredia. Definoval som si v ňom však vlastné parametre, ktoré sú následne v aplikácii využité.
 - `document_directory` – Adresár, kde majú byť ukladané nahrané dokumenty.
 - `document_backup` – Adresár zálohy dokumentov.
 - `database_backup` – Adresár zálohy databázy.
 - `backup_delete` – Tento parameter určuje interval dní, po ktorom má byť záloha odstránená.

Po nastavení týchto konfiguračných údajov je možné systém inicializovať príkazom definovaným v `/InitSystemCommand.php`.

6.2.1 Príkazy cron

Pre správne sledovanie zmien v záznamoch v obchodnom registri a ukladanie, či mazanie dokumentov a stavov databázy je nevyhnutné nakonfigurovať plánovač úloh Cron⁴. Je možné zavolať príkaz `"crontab -e"`, ktorý otvorí crontab práve prihláseného užívateľa. Crontab je súbor, do ktorého je možné pridávať príkazy s údajmi špecifikujúcimi čas spustenia. Cron následne v čase očakávaného spustenia vykoná zadaný príkaz.[\[25\]](#)[\[26\]](#)

⁴Cron je plánovač úloh pre mnoho operačných systémov.

Mnou vytvorené záznamy v tomto súbore sú vo formáte: `"* * * * * php $PATH_TO_LAWOFFICE/bin/console príkaz".` Podľa číselného nastavenia hviezdíčiek sa určí čas vykonania príkazu. Vysvetlenie k hviezdíčkam podľa poradia je nasledovné:[26]

1. Nastavenie minút.
2. Nastavenie hodín.
3. Nastavenie dňa v mesiaci.
4. Nastavenie mesiaca.
5. Nastavenie dňa v týždni.

Druhou časťou záznamu je len štandardné spúšťanie príkazov Symfony⁵. Príkazy, ktoré je pre správny beh systému nutné pridať do konfigurácie sú tieto:

- `app:check-records` Kontrola zmien sledovaných záznamov v obchodnom registri.
- `app:backup-create` Vytvorenie zálohy.
- `app:backup-delete` Odstránenie zálohy.

6.2.2 Inicializácia systému

System je možné inicializovať do počiatočného stavu, čiže skoro prázdnej databázy. Táto databáza obsahuje iba záznamy o typoch zamestnancov, dokumentov a úkonov a záznam vedúceho kancelárie.

Príkaz sa spúšťa rovnako, ako príkazy uvedené v podkapitole venujúcej sa konfigurácii plánovača úloh Cron. Príkaz vyzerá nasledovne: `"app:init-system login_vedúceho heslo_vedúceho".` Podľa dvoch zadaných parametrov sa vedúcemu kancelárie nastaví meno a heslo a tým mu umožní prístup do systému.

6.3 Webové užívateľské rozhranie systému

Informačný systém pre advokátske kancelárie bude využívaný užívateľmi, ktorí nemusia mať hlboké znalosti práce s počítačom. Preto som sa pri jeho návrhu a implementácii snažil o jednoduché a ľahko pochopiteľné užívateľské rozhranie.

Na implementáciu grafického užívateľského rozhrania som využíval HTML, CSS a JavaScript.

6.3.1 Navigácia

Kostru užívateľského rozhrania som budoval na základe Bootstrap⁶ témy "SB Admin"⁷. Aplikácia je teda vďaka použitiu frameworku Bootstrap zobraziteľná aj na mobilných zariadeniach. Navigácia je rozdelená na tri hlavné časti:

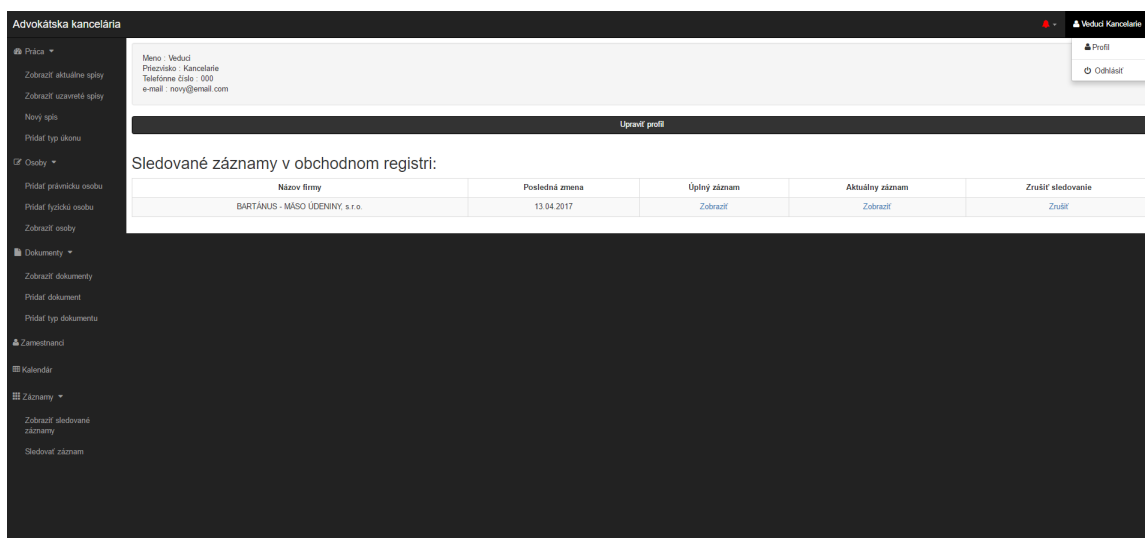
⁵Spôsob je možné nájsť tu <http://symfony.com/doc/current/console.html>

⁶Bootstrap je HTML, CSS a JS framework[8]

⁷odkaz: <https://startbootstrap.com/template-overviews/sb-admin/>

- Bočný panel – Bočný panel umožňuje prechod medzi hlavnými časťami implementovaného systému. Obsahuje šesť hlavných kariet, z ktorých štyri sú rozvinuteľné, čím umožňujú prístup ku konkrétnejším akciám v rámci danej časti systému. Pokiaľ užívateľ načíta stránku z určitej sekcie, karta tej sekcie bude zvýraznená. Usporiadanie položiek, ktoré sú zahrnuté v rozvinuteľných kartách som zvolil podľa odhadovanej pravdepodobnosti prístupu k daným položkám.
- Horný panel – Na hornom paneli sa nachádzajú dve významné rozvinuteľné položky. Jednou z nich je meno užívateľa, pomocou ktorého má užívateľ prístup k vlastnému profilu a možnosti odhlásenia sa zo systému. Druhou položkou je znak zvonu, ktorý slúži na signalizáciu nových oznámení a prístup k nim. Po načítaní každej stránky sa zašle na server AJAX žiadosť o počet nových oznámení, ktoré momentálne prihlásený užívateľ má. V prípade, že má užívateľ nové oznámenia, znak zvonu sčervenie. Táto položka po rozvnutí zobrazí počet nových oznámení a spomínanú možnosť ich zobrazenia.
- Informačná plocha – Táto plocha slúži na zobrazenie dát systému a manipuláciu s nimi. Zvolil som zobrazenie údajov na celú šírku tejto plochy, kvôli prehľadnosti zobrazenia. Ak sa v tejto ploche nachádza samostatné tlačidlo, je taktiež zobrazené na celú šírku plochy. V opačnom prípade sú tlačidlá pravidelne usporiadané.

Jednotlivé prvky navigácie je možné vidieť na obrázku 6.1



Obr. 6.1: Navigačné prvky informačného systému pre advokátske kancelárie

6.3.2 Zobrazenia zoznamov údajov

Je zrejmé, že informačný systém musí zobrazovať zamestnancovi sadu rozličných údajov. V prípade potreby ukázať užívateľovi skupinu údajov rovnakého typu je vhodné túto skupinu usporiadať do prehľadnej podoby. V mojom systéme som sa rozhodol pre jednoduché usporiadanie do tabuliek.

Zoznamy požadovaných údajov⁸ sú preto organizované v prehľadných tabuľkách. Vďaka tejto organizácii je možné s jednotlivými záznamami jednoducho manipulovať.

Možnosti manipulácie so záznamami v tabuľkách podľa typu záznamov môžu byť:

- Spisy – Zobrazíť detail spisu.
- Úkony – Zobrazíť detail úkonu, zmazať úkon.
- Osoby – Odstrániť osobu, pridať klienta/pridať protistranu.
- Dokument – Zobrazíť detail dokumentu.
- Žiadosti o dokument – Pridať nový dokument, pridať existujúci dokument, odobrať žiadosť od úkonu, stiahnuť dokument.(Chápané z hľadiska systému.)
- Zamestnanci – Prijatť späť/prepustiť zamestnanca, oddeliť zamestnanca od spisu.
- Záznamy – Zobrazíť aktuálny záznam v ochodnom registri, zobrazíť úplný záznam, zrušiť/započať sledovanie.

Po započatí manipulácie s vybraným záznamom v tabuľke sa načíta URL vygenerované pomocou funkcie "path"frameworku Twig. Toto URL v sebe zvyčajne obsahuje identifikátor objektu, s ktorým je manipulované. Controller, ktorý zodpovedá tomuto URL si z neho následne dokáže potrebný identifikátor extrahovať a vykonať akciu zodpovedajúcu vnútornej logike controller-u.

Príklady niektorých zobrazení zoznamov je možné vidieť ako súčasť detailov v obrázkoch 6.3 a 6.4.

Špeciálnymi prípadmi zoznamov údajov sú zobrazenie kalendáru lehôt a užívateľských upozornení. Kalendár lehôt je síce usporiadaný v tabuľke znázorňujúcej momentálne zobrazený mesiac, nie je však možné s jednotlivými lehotami manipulovať. Lehoty sa do tohto kalendáru načítavajú podobným spôsobom ako sledovanie počtu nových oznámení. Užívateľ po načítaní stránky zašle AJAX žiadosť a ako odpoveď sa vráti súbor vo formáte JSON, z ktorého kalendár extrahuje údaje na zobrazenie lehôt. Udalosti týkajúce sa práce sa zobrazujú čiernou farbou a udalosti vytvorené užívateľom sú zobrazené bielou. Na implementáciu kalendáru som využil knižnicu "FullCalendar"⁹ implementovanú v jazyku JavaScript. Na implementáciu vznášajúcich sa poznámok pre jednotlivé záznamy v kalendári som využil knižnicu qTip².¹⁰

Príklad zobrazenia kalendára je možné vidieť na obrázku 6.2

Užívateľské upozornenia sú unikátne svojím spôsobom zobrazenia. Upozornenia sú totiž zobrazené vedľa seba a sú farebne rozlíšené podľa typu upozornenia. Rozlišujú sa upozornenia týkajúce sa spisov(modré) a upozornenia týkajúce sa sledovaných záznamov(zelené). Takýto spôsob zobrazenia som zvolil kvôli rozsiahlosti obsahu upozornení a ich príjemnejšiemu čítaniu.

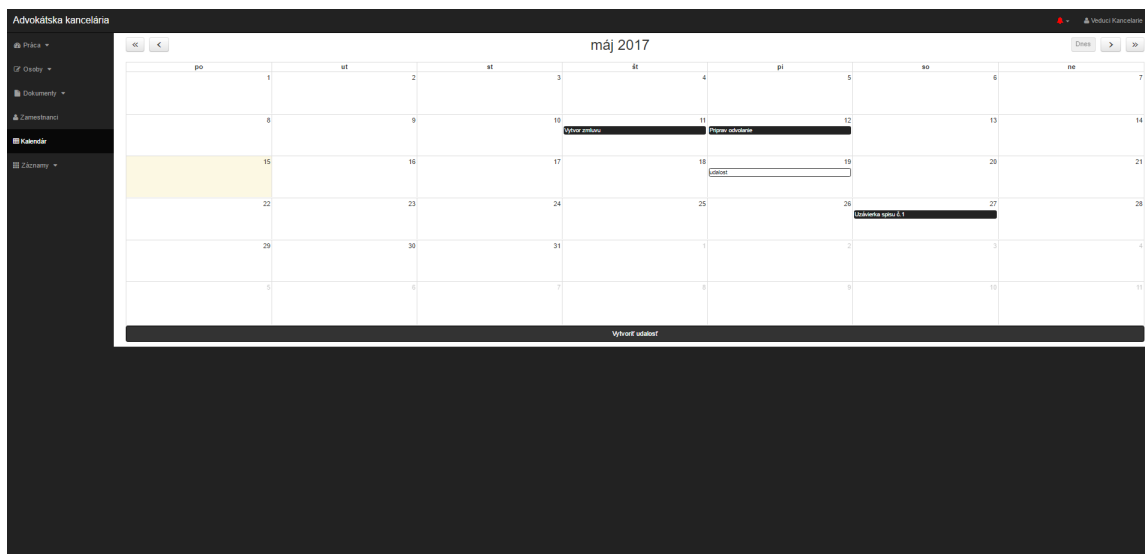
6.3.3 Detaily

Po prístupe ku detailu konkrétneho záznamu v tabuľke je potrebné zobrazíť relevantné údaje k tomuto záznamu, prípadne ďalšie položky v systéme, ktoré majú s daným záznamom

⁸Spisov, osôb, záznamov, sledovaných záznamov, zamestnancov, úkonov, dokumentov a žiadostí o dokumenty.

⁹odkaz: <https://fullcalendar.io/>

¹⁰odkaz: <http://qtip2.com/>



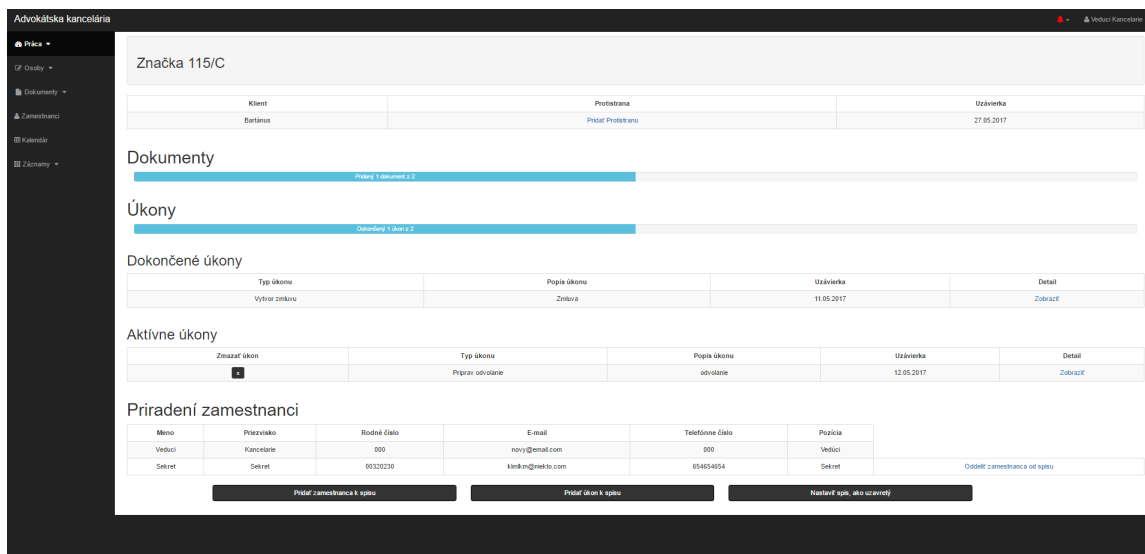
Obr. 6.2: Obrazovka kalendáru lehôt.

určitý vzťah. Taktiež je v detailoch záznamov umožnené vykonávať špecifickejšie operácie nad záznamami, ktorých detaily sú zobrazené.

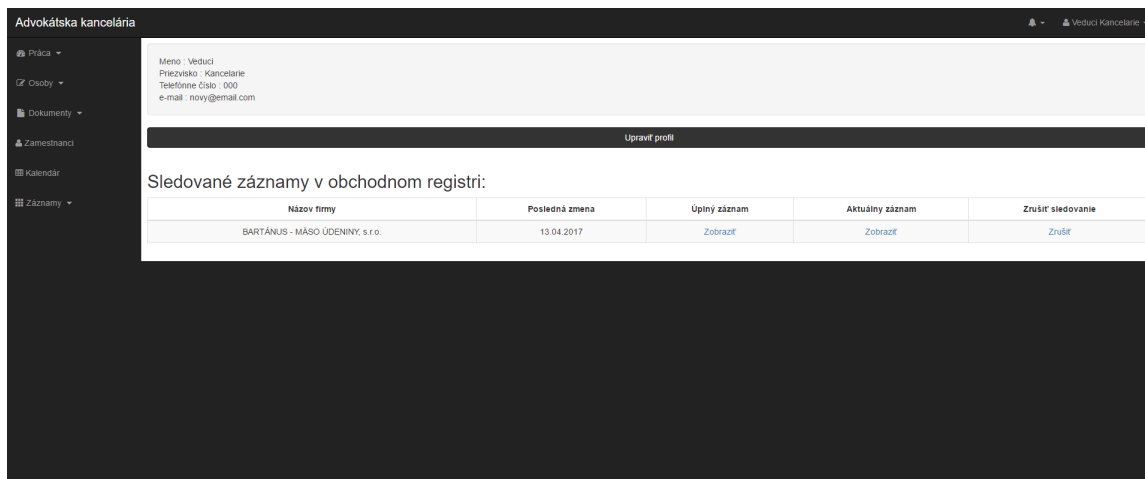
V implementovanej aplikácii je možné zobraziť detaily týchto položiek:

- Spis – Detail spisu je najdôležitejšou súčasťou systému z hľadiska organizácie práce. Sú v ňom zobrazené základné informácie viažuce sa na spis ako poznámka, klient, protistrana a uzávierka, no aj zoznam úkonov vyžadovaných k dokončeniu spisu a zamestnanci, ktorí sa na vypracovaní spisu majú podieľať. Úkony môžu byť dokončené a nedokončené. Od pomeru počtu dokončených úkonov a celkového počtu úkonov vyžadovaných v rámci spisu sa odvíja úroveň naplnenia jedného z dvoch indikátorov dokončenia práce. Naplnenie druhého indikátoru sa odvíja od pomeru počtu dokumentov pridaných k žiadostiam o dokumenty v rámci jednotlivých úkonov spisu a celkového súčtu takýchto žiadostí. V rámci spisu je taktiež možné, okrem opísanej manipulácie so záznamami v zoznamoch, pridať zamestnanca k spisu, pridať nový úkon k spisu, či označiť spis ako uzavretý, alebo aktívny.
- Úkon – Detail úkonu obsahuje, rovnako ako spis, poznámku a dátum uzávierky. Okrem toho obsahuje zoznam žiadostí o dokument, ku ktorým môže byť potrebný dokument priradený. Podobne ako spis obsahuje indikátor priradených dokumentov. Z detailu úkonu je možné sa vrátiť na spis, ku ktorému je priradený. Taktiež je možné pridať novú žiadosť o dokument, či nastaviť úkon ako ukončený/aktívny.
- Dokument – Detail dokumentu obsahuje okrem informácií ohľadom dokumentu aj spisy, v rámci ktorých je priradený ku žiadostiam o dokument. Je možné tento dokument stiahnuť do počítača a vymazať, pokiaľ k žiadnym spisom priradený nie je.
- Profil užívateľa – Profil užívateľa obsahuje údaje momentálne prihláseného užívateľa, ktoré je možné upraviť a záznamy v obchodnom registri, ktoré momentálne sleduje.

Príklady zobrazenia týchto detailov je možné si prehliadnúť v obrázkoch 6.3 6.4.



Obr. 6.3: Detail spisu.



Obr. 6.4: Detail užívateľa.

6.3.4 Formuláre

Keďže je potrebné informačný systém plniť údajmi, je nutné implementovať spôsob akým sa toho dá dosiahnuť. Na takýto účel sú ideálne formuláre, ktoré môže užívateľ vyplniť a tým zaslať systému štruktúrované údaje.

Vytvoril som ich hneď niekoľko:

- Vytvoriť spis.
- Vytvoriť fyzickú/právnicku osobu.
- Vytvoriť dokument.
- Vytvoriť typ dokumentu.
- Vytvoriť typ úkonu.

- Prijat' nového zamestnanca.
- Vytvoriť udalosť v kalendári.
- Upraviť svoj profil.
- Vytvoriť úkon.
- Vytvoriť žiadosť o dokument.
- Vyhľadať záznam v obchodnom registri Slovenskej republiky.
- Filtrovať dokumenty.

Všetky tieto formuláre, okrem posledných dvoch slúžia na vytvorenie nejakého záznamu v databáze. Vyhľadávanie v obchodnom registri je bližšie popísané v podkapitole 6.5. Filtrovanie dokumentov slúži na získavanie výberu dokumentov z databázy na základe vybraných kritérií. Je možné vyhľadať dokumenty pridané len určitými zamestnancami, či dokumenty konkrétneho typu. Taktiež je možné vybrať rozsah dátumov, počas ktorého boli zobrazené dokumenty nahraté do systému. Posledným parametrom vyhľadávania je priradenosť dokumentu ku spisu, čo ovplyvňuje možnosť ich odstránenia zo systému.

Napriek tomu, že je prihlasovanie užívateľa taktiež formulár, na jeho vytvorenie som nepoužil objekt typu `FormBuilderInterface`. Vyhotovil som ho pomocou návodu uvedeného v dokumentácii symfony.[19]

Na zobrazenie prehľadného kalendára pri výbere dátumov, ktoré sú vyžadované v niektorých poliach formulárov som využil knižnicu `bootstrap-datepicker`¹¹.

Príklady formulárov je možné vidieť na obrázkoch 6.5 a 6.6.

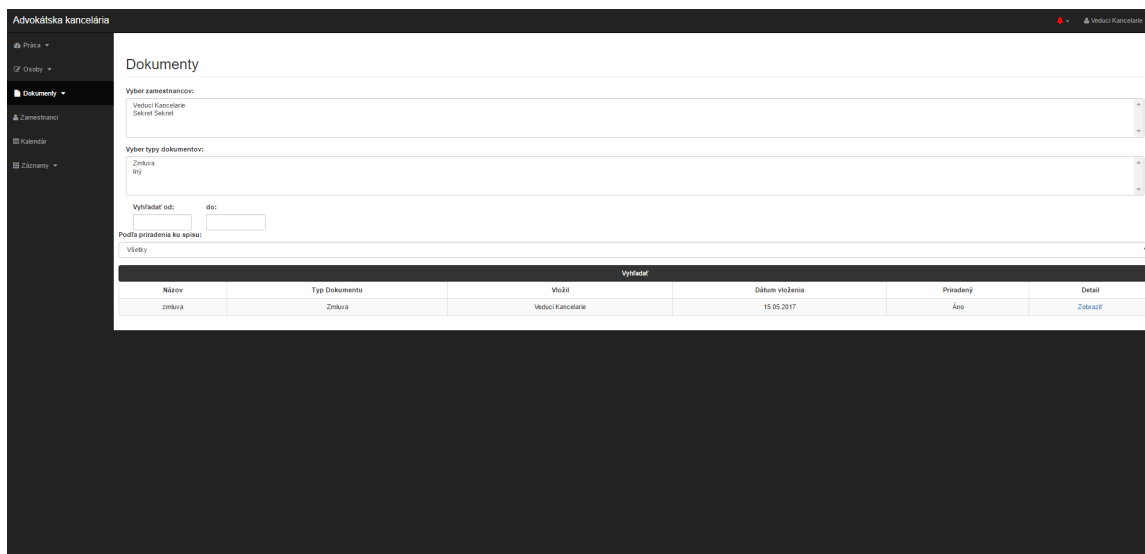
The screenshot shows a web interface for a law firm's office management system. The main content area contains a form for creating an act (úkon). The form has several sections:

- Zvoľte typ zložky práce:** A dropdown menu with 'Vytvor zmluvu' selected.
- Poznámky k práci:** A text input field.
- Dátum uzávierky:** A date picker field.
- Požadované dokumenty:** A section with a text input field for 'Popis dokumentu:' and a dropdown menu for 'Typ dokumentu:' with 'Zmluva' selected.
- Buttons:** 'Uložiť a ukončiť' and 'Ukončiť a prejsť ďalší úkon'.

 The left sidebar contains navigation links: 'Práca', 'Osoby', 'Dokumenty', 'Zamestnanci', 'Kalendár', and 'Záznamy'.

Obr. 6.5: Formulár tvorby úkonu.

¹¹odkaz: <https://bootstrap-datepicker.readthedocs.io/en/latest/>



Obr. 6.6: Filtrácia dokumentov.

6.4 Databázový systém

Práca s databázou je dôležitá pre uchovávanie perzistentných dát v rámci systému. Na vytvorenie základnej kostry databázy v systéme pre riadenie bázy dát MySQL som využil jednoduchý nástroj Adminer¹².

Na zjednodušenie databázových operácií počas vývoja aplikácie som sa obrátil na ORM a DBAL¹³ na ktorom je postavený. Oba tieto nástroje sú vyvinuté v rámci projektu Doctrine. Tieto nástroje ponúkajú elegantný spôsob abstrakcie tabuliek v databáze a operácií nad databázou s využitím princípov OOP¹⁴.^[18]

6.4.1 Štruktúra databázy

Pri budovaní databázy som sa sústredil na reprodukciu ER modelu uvedeného v kapitole 5 do podoby MySQL databázy.

Okrem entít reprezentovaných v ERD bolo nutné rozšíriť databázu o tabuľky uchovávajúce údaje o lehotách a oznámeniach. Taktiež bolo potrebné zhotoviť pomocné tabuľky reprezentujúce M:N vzťahy medzi entitami. Výsledná databáza je zobrazená na obrázku 6.7.

Po úspešnom vytvorení databázy som potreboval získať ich objektovú reprezentáciu v rámci aplikácie. Na generáciu tried reprezentujúcich tabuľky v databáze som využil nasledujúce príkazy:^[19]

1. `"php bin/console doctrine:mapping:import -force AppBundle annotation"`
2. `"php bin/console doctrine:generate:entities AppBundle"`

Prvý príkaz slúži na generovanie tried, reprezentujúcich údaje. Po vykonaní druhého príkazu sa vnútri tried vygenerujú metódy umožňujúce manipuláciu s objektami v rámci

¹²odkaz: <https://www.adminer.org/>

¹³DBAL – Database abstraction layer

¹⁴OOP – Object oriented programming

aplikácie. Údaje týchto objektov budú premietnuté do záznamov v databáze spôsobom popísaným v podkapitole 6.4.2.

Dokument
Dokument_to_ukon
Event
Event_to_Zamestnanec
Osoba
Oznamenie
Oznamenie_to_zamestnanec
Spis
Spis_to_zamestnanec
Typ_Dokumentu
Typ_Ukonu
Typ_zamestnanca
Ukon
Zamestnanec
Zaznam
Zaznam_to_zamestnanec
Ziadany_Dokument

Obr. 6.7: Výsledná databáza.

6.4.2 Práca s databázou

Operácie s databázou, ako napríklad vkladanie údajov do databázy a ich získavanie, sú vďaka Doctrine jednoduché a bezpečné.

Vkladanie údajov do databázy je realizované pomocou objektu `EntityManager`¹⁵ nasledovnými krokmi:[19]

1. Vytvorenie inštancie triedy reprezentujúcej korešpondujúcu tabuľku v databáze.
2. Naplnenie výsledného objektu potrebnými údajmi.
3. Zachovanie objektu v rámci objektu `EntityManager`.
4. Vykonanie SQL príkazu na vloženie údajov objektu do záznamu v databáze.

Objekt `EntityManager` nám poskytuje jednoduché metódy na vykonanie posledných dvoch krokov. Na zachovanie objektu v rámci objektu `EntityManager` je využitá metóda `persist(vkladany_objekt)`. Vykonanie SQL príkazu je zabezpečené metódou `flush()`. [12]

Získavanie údajov z databázy sa realizuje buď nad celým objektom `EntityManager`, alebo nad repozitármi jednotlivých entít. Na získavanie objektov a ich údajov z databázy sú využiteľné tri spôsoby.[19]

- Jednoduché príkazy nad repozitármi – Tento spôsob je ideálny pri vykonávaní jednoduchých požiadavok. Je možné získať jeden, či viacero údajov v závislosti od použitej metódy repozitára entít. Použiteľné metódy sú vygenerované podľa stĺpcov tabuliek a je teda možné v databáze vyhľadávať podľa hodnôt záznamov v týchto stĺpcoch. Je taktiež možné získať všetky záznamy z daného repozitára.
- Využívanie DQL¹⁶ – DQL je jazyk vyvinutý v rámci projektu Doctrine a je podobný klasickému SQL. Budovanie zložitejších príkazov na získavanie údajov z databázy je

¹⁵Tento objekt je súčasťou projektu Doctrine

¹⁶DQL – Doctrine query language

použitím tohto jazyka zjednodušené a bezpečné, pokiaľ je príkaz vybudovaný správnym spôsobom. Nebezpečenstvo nastáva pri vytváraní príkazu konkatenáciou reťazcov obsahujúcich príkazy DQL s užívateľským vstupom. Spôsob, akým sa dá predísť nebezpečenstvu vloženia nežiaduceho SQL príkazu pri zadávaní údajov vkladaných užívateľom je využitie parametrov v rámci DQL požiadavku. Doctrine vtedy totiž automaticky sanitizuje užívateľský vstup.

- Využívanie objektu `QueryBuilder` – Tento objekt, využívajúci návrhový vzor `Builder`, dokáže pomocou jeho metód vyskladať DQL príkaz. Je vhodný na budovanie dynamických žiadostí.

Všetky tri spôsoby som v mojom projekte využil. Zaujímavé využitie pre objekt `QueryBuilder` som našiel pri filtrácii dokumentov podľa ľubovoľných parametrov.

6.5 Napojenie na obchodný register

Vyhľadávanie v obchodnom registri na internete je jednoduché a poskytuje možnosť vyhľadávania použitím rôznych parametrov. Ja som sa rozhodol implementovať vyhľadávanie podľa IČO, kvôli jednoduchosti, jednoznačnosti a pravidelnému používaniu v rámci advokátskej kancelárie.

6.5.1 Vyhľadanie záznamu

Vyhľadávanie záznamu môže užívateľ v systéme vykonať dvomi spôsobmi. Prvým je vyhľadanie záznamu týkajúceho sa osoby, ktorá sa v systéme nenachádza zadaním jej IČO do vyhľadávacieho poľa. Druhým je vyhľadávanie osoby nachádzajúcej sa v databáze podľa IČO, ktoré jej bolo priradené.

Oba tieto spôsoby využívajú metódu triedy `findByIco(IČO)` triedy `orsrFinder`. Táto metóda zostaví štandardné URL generované pri vyhľadávaní v obchodnom registri a nahrá HTML obsah stránky do lokálnej premennej.

Nájdených záznamov môže byť niekoľko¹⁷. Je teda potrebné rozložiť obsah stránky na relevantné údaje a tie užívateľovi zobraziť. Na tento účel využívam komponent projektu `Symfony` s názvom `DomCrawler`. Tento nástroj umožňuje jednoduchú navigáciu v rámci HTML kódu na základe uzlov.[\[19\]](#)

V spomínanej metóde si odfiltrujem názvy firiem, ktoré sa nachádzajú v značkách `<div>` s atribútom `"sbj"`. Potom extrahujem URL aktuálnych a úplných záznamov o právnickej osobe. Vytvorím objekty typu `záznam`, ktoré je možné po zobrazení zoznamu nájdených záznamov označiť na sledovanie. Záznamy, ktoré neboli označené na sledovanie sa zo systému odstraňujú.

6.5.2 Kontrola zmien záznamov

Pri označení záznamu na sledovanie sa do objektu pomocou metódy `findLatestChange(aktuálny link)` triedy `orsrFinder` naplní dátum poslednej zmeny záznamu. Činnosť získavania poslednej zmeny je vykonávaná podobným spôsobom ako vyhľadávanie záznamov v obchodnom registri, s tým rozdielom, že URL HTML stránky, ktorej obsah extrahujem, mám už uložené v databáze. Načítam si teda obsah stránky obsahujúcej

¹⁷Napríklad v prípade zmeny sídla firmy a následnom zapísaní do obchodného registra iného súdu.

aktuálny záznam v obchodnom registri a vyfiltrujem značky `` s atribútmi "ro"¹⁸ a "ra". Potom nájdem tie výsledky filtru, ktoré obsahujú dátum zmeny. Tieto dátumy porovnam a vložím ten najvyšší do položky označujúcej dátum zmeny.

Na kontrolu potenciálnej zmeny záznamu sa využíva rovnaká metóda, s tým rozdielom, že jej výsledok do záznamu nevkladám, pokiaľ nie je vyšší, ako už vložený dátum. Pokiaľ vyšší je, užívateľ sledujúci daný záznam dostane upozornenie o zmene záznamu.

Pre konfiguráciu pravidelnej kontroly záznamov viď kapitolu 6.2.

6.6 Súborový systém

Ukladanie dokumentov potrebných pre výkon advokácie je jednou zo základných požiadavok môjho systému. Tieto dokumenty a databázu je taktiež potrebné zachovať v podobe zálohy. Koreňové adresáre, určené na ukladanie adresárov obsahujúcich tieto súbory, je možné nastaviť podľa návodu uvedenom v podkapitole 6.2.

6.6.1 Dokumenty

Adresárová štruktúra dokumentov v mojom systéme je jednoduchá. Dokumenty sa ukladajú s názvom vygenerovaným systémom a to v takomto formáte: "*(typ dokumentu)*_*(názov dokumentu zadaný užívateľom)*_*(dátum vloženia)*_*(identifikátor dokumentu)*_*(predpokladaná prípona podľa obsahu dokumentu)*".

Súbor s takýmto názvom sa následne uloží do adresára `/vsetky_dokumenty/`. Pokiaľ je dokument priradený ku spisu, bude naň vytvorený symbolický odkaz v novo vytvorenom adresári, ktorý nasleduje formát: `/(Priezvisko/Názov firmy klienta)/(identifikátor spisu)/`.

Pokiaľ sa užívateľ rozhodne dokument od spisu odlúčiť, symbolický odkaz bude odstránený. Odstránenie dokumentu zo systému a tým aj zo zložky `/vsetky_dokumenty/` je možné len pokiaľ nie je dokument priradený ku žiadnemu spisu.

6.6.2 Záloha

Jednoduchý zálohovací systém som zabezpečil generovaním adresárov, a databázového súboru, ktorých názov je časová značka. Tento prístup umožňuje jednoduché obnovenie systému zo zálohy pomocou príkazu `php $PATH_TO_LAWOFFICE/bin/console app:system-restore počet_dni`. Parameter tohto príkazu určí zálohu, z ktorej sa má systém obnoviť. Dátum zálohy sa vypočíta pomocou získania rozdielu dnešného dátumu a počtu dní, ktorý užívateľ zadal ako parameter.

Databáza sa ukladá jednoducho a to vygenerovaním súboru, ktorý obsahuje sadu SQL príkazov, ktorými je možné reprodukovat vtedajší stav databázy. Na tento účel využívam príkaz `mysqldump`.^[11]

Záloha dokumentov prebieha skopírovaním celého obsahu koreňového adresára úložiska dokumentov, do adresára vygenerovaného počas vykonávania zálohy.

6.7 Testovanie

Pred uvedením informačného systému do prevádzky je veľmi dôležité ho správne otestovať. Testovaním systému je možné vyhnúť sa neočakávaným chybám systému, ktoré by mohli

¹⁸tento atribút sa využíva na zvýraznenie zmeny položky záznamu v obchodnom registri.

počas jeho používania spôsobiť nežiaduce komplikácie. Test zobrazenia užívateľského rozhrania som realizoval v prehliadači Google Chrome¹⁹ za použitia bežného zobrazenia a zobrazenia na dotykovom zariadení.

6.7.1 Jednotkové testovanie

Jednotkové testovanie je spôsob testovania základnej funkcionality vybranej triedy.[19] Je často využívané na testovanie istého bloku kódu. Tento typ testovania som realizoval pomocou testovacieho frameworku PHPUnit²⁰.

Tento spôsob testovania som využil na preverenie správneho pracovania mojich Services. Zvlášť užitočné sa ukázalo byť testovanie správnej funkcionality vyhľadávača záznamov a ich zmien v obchodnom registri. Pomohlo mi odhaliť nečakanú chybu pri triedení HTML kódu za účelom získania najvyššieho dátumu na stránke. V prípade, že bol záznam pozmenený, značka vymedzujúca dátum zmeny môže obsahovať alternatívny atribút. Túto chybu som následne opravil.

6.7.2 Testovanie funkcionality

Testovanie správnej funkcionality systému som realizoval naplnením testovacej databázy, ktorá je využívaná v niektorých jednotkových testoch, priamo cez nástroje vyvinutej aplikácie. Tým som overil správanie formulárov, či vytváranie vzťahov medzi jednotlivými záznamami v databáze.

Tento spôsob testovania aplikácie mi pomohol nájsť niekoľko drobných chýb a doladiť prehliadnuté implementačné detaily. Príkladmi nájdených závad sú napríklad chýbajúce odstraňovanie lehôt po zmazení úkonu, alebo nenaprogramované odstránenie sledovaného záznamu z databázy, pokiaľ ho už žiaden zo zamestnancov nesleduje.

Na nájdenie zvyšných nedostatkov programu plánujem využiť budúcich užívateľov aplikácie. Pred ostrým zavedením do prevádzky budú užívatelia upozornení na skúšobné obdobie systému, počas ktorého nemajú záruku predpokladanej finálnej funkcionality a teda by mali svoje dokumenty a údaje ukladať aj iným spôsobom.

Počas tejto skúšobnej doby budem získavať odozvu od užívateľov ohľadom spokojnosti so systémom, či prípadných chýb. Na jej konci, po oprave všetkých nájdených chýb, systém oficiálne zavediem do prevádzky a umožním jeho používanie.

6.8 Zhrnutie

V tejto kapitole bol opísaný spôsob implementácie výslednej webovej aplikácie a jednotlivé nástroje, ktoré som sa pri vývoji systému rozhodol využiť.

Výsledkom je webová aplikácia, do ktorej bude po zavedení možné prístupíť cez internet, kdekoľvek sa užívateľ nachádza. Pokiaľ to bude advokátska kancelária vyžadovať, je samozrejme možné túto aplikáciu uviesť do prevádzky len v rámci lokálnej siete kancelárie, čím sa môže zvýšiť zabezpečenie systému.

Nástroje, ako kalkulačky trovov, či cestovných nákladov, neboli implementované.

¹⁹odkaz: <https://phpunit.de/>

²⁰odkaz: <https://www.google.com/chrome/>

Kapitola 7

Záver

Zadaním bakalárskej práce bolo vyvinúť informačný systém pre advokátske kancelárie, umožňujúci spracovávanie dokumentov a organizáciu práce zamestnancov advokátskej kancelárie. Prieskum súčasného stavu advokácie na území Slovenskej republiky mi poskytol nápady na vhodný spôsob realizácie tejto úlohy. Systém som preto rozšíril aj o možnosť sledovania záznamov v obchodnom registri na internete. Okrem snahy uľahčenia práce advokátom, ktorí budú môj systém používať, som sa zameral aj na zabezpečenie údajov a dokumentov, ktoré v systéme môžu byť uchovávané.

Možné súčasti aplikácie boli konzultované s advokátmi JUDr. Jánom Bartánusom a JUDr. Miloslavom Hrickom. Ich poznatky z oboru advokácie mi počas celého priebehu práce na aplikácii formovali názor na to, aké nástroje by mala výsledná aplikácia obsahovať. Taktiež mi poskytli prehľad o súčasne platnej legislatíve ohľadom informačných systémov, ktoré narábajú s citlivými údajmi.

Aby som bol schopný implementovať výsledný software, musel som sa naučiť veľa poznatkov o budovaní webových stránok a databáz. Výsledkom mojej snahy je funkčný informačný systém, ktorý je reálne využiteľný v advokátskych kanceláriách rôzneho typu. Verím, že jeho grafické užívateľské rozhranie bude ľahko pochopiteľné pre väčšinu užívateľov. Neoceniteľný návod na správny prístup k implementácii a využitie rôznych užitočných nástrojov mi ponúkli oficiálne dokumentácie projektov Symfony a Doctrine. Práve nasledovanie bežných praktík pri vývoji Symfony aplikácie poskytuje v budúcnosti možnosť jednoduchého rozšírenia funkcionality.

Aplikácia bola testovaná viacerými spôsobmi. Každý z nich mi umožnil nájsť chyby systému, ktoré sa mi podarilo odstrániť. Napriek tomu, že môj prístup k testovaniu odhalil niekoľko chýb, verím, že pred zavedením do skutočnej kancelárie je potrebné systém otestovať dôkladnejšie.

Analýza software-ových nástrojov pre advokátsku prax mi poskytla obraz o tom, ako by mohla vyzeráť profesionálne vyvinutá aplikácia. Software-ové riešenia spomínané v kapitole poskytujú niekoľko užitočných nástrojov, o ktoré by bolo vhodné moju aplikáciu rozšíriť. Príkladom takýchto nástrojov je napríklad výpočet fakturácie klienta, či napojenie priamo na obchodný vestník. Taktiež by bolo zaujímavé do aplikácie zahrnúť výpočet cestovných nákladov s napojením na Google Maps API. Podobným spôsobom by bolo možné rozšíriť kalendár lehôt o údaje užívateľa z jeho osobného Google Calendar-u.

Literatúra

- [1] Zákon č. 122/2013 Z. z. o ochrane osobných údajov.
- [2] Zákon č.586/2003 Z. z. o advokácii.
- [3] Vyhláška č. 164/2013 Z. z. o ochrane osobných údajov.
- [4] Vyhláška 655/2004 Z. z. o odmenách a náhradách advokátov.
- [5] Zákon č. 540/2001 Z.z. o štátnej štatistike (§ 2f).
- [6] *AdvocatusDigital, právny software na správu advokátskej kancelárie* . [Online; navštívené 11.05.2017].
URL <http://www.advocatusdigital.com/>
- [7] *ADVOS, komplexný informačný systém určený pre prácu v advokátskej kancelárii*. [Online; navštívené 11.05.2017].
URL <http://www.advos.sk/>
- [8] *Bootstrap*. [Online; navštívené 15.05.2017].
URL <https://v4-alpha.getbootstrap.com/>
- [9] *Definícia právnych pojmov, oficiálna stránka Ministerstva spravodlivosti Slovenskej republiky* . [Online; navštívené 15.05.2017].
URL <https://www.justice.gov.sk/Stranky/Sudy/Sudne-rozhodnutia/Vysvetlivky.aspx>
- [10] *Diagram prípadov použitia*. [Online; navštívené 12.05.2017].
URL https://sk.wikipedia.org/wiki/Diagram_pr%C3%ADpadov_pou%C5%BEitia
- [11] *Dokumentácia k príkazu mysqldump*. [Online; navštívené 16.05.2017].
URL <https://dev.mysql.com/doc/refman/5.7/en/mysqldump.html>
- [12] *Dokumentácia projektu Doctrine*. [Online; navštívené 15.05.2017].
URL <http://www.doctrine-project.org/>
- [13] *Entitno-relačný diagram*. [Online; navštívené 12.05.2017].
URL https://cs.wikipedia.org/wiki/Entity-relationship_model
- [14] *LEGUS 2.0, právny systém*. [Online; navštívené 11.05.2017].
URL <https://www.legus.sk/>
- [15] *Model-View-Controller wikibooks stránka*. [Online; navštívené 14.05.2017].
URL https://en.wikibooks.org/wiki/Computer_Science_Design_Patterns/Model%E2%80%93view%E2%80%93controller

- [16] *Obchodný register Slovenskej republiky*. [Online; navštívené 10.05.2017].
URL <http://www.orsr.sk>
- [17] *Sekvenčný diagram*. [Online; navštívené 12.05.2017].
URL https://sk.wikipedia.org/wiki/Sekven%C4%8Dn%C3%BD_diagram
- [18] *Stránka projektu Doctrine* . [Online; navštívené 15.05.2017].
URL <http://www.doctrine-project.org/>
- [19] *Symfony dokumentácia*. [Online; navštívené 13.05.2017].
URL <http://symfony.com/doc/>
- [20] *Symfony oficiálna stránka*. [Online; navštívené 13.05.2017].
URL <http://symfony.com/>
- [21] *Tribunal, software-ové riešenie administratívy advokátskej kancelárie*. [Online; navštívené 11.05.2017].
URL <http://www.tribunal.wai.sk/>
- [22] Arlow, J.; Neustadt, I.: *UML 2 a unifikovaný proces vývoja aplikácií*. Brno: Computer Press, a.s., 2007, ISBN 978-80-251-1503-9.
- [23] Marek, M.: *Obchodný register SR na internete*. August 2000, [Online; navštívené 10.05.2017].
URL <http://mobilmania.azet.sk/clanok/73477/obchodny-register-sr-na-internete>
- [24] Provos, N.; Mazières, D.: *A Future-Adaptable Password Scheme*. In *Proceedings of 1999 USENIX Annual Technical Conference*, Apríl 1999, str. 81–92.
URL https://www.usenix.org/legacy/events/usenix99/provos/provos_html/node1.html
- [25] Vixie, P.: *cron(1) Linux User's Manual*. Apríl 2010.
- [26] Vixie, P.: *crontab Linux User's Manual*. Apríl 2010.

Príloha A

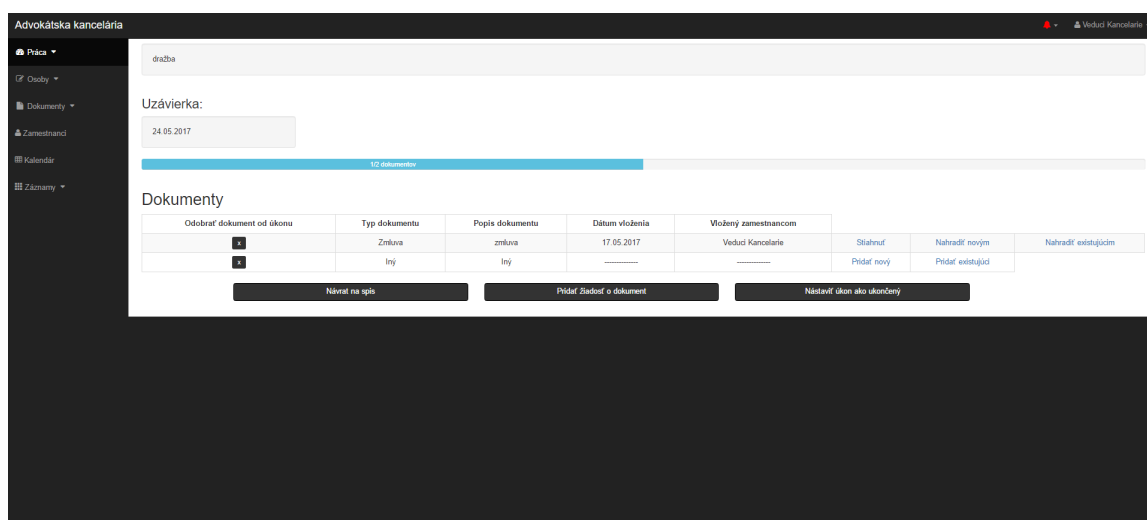
Obsah priloženého média

Priložené médium obsahuje nasledovné adresáre a súbory:

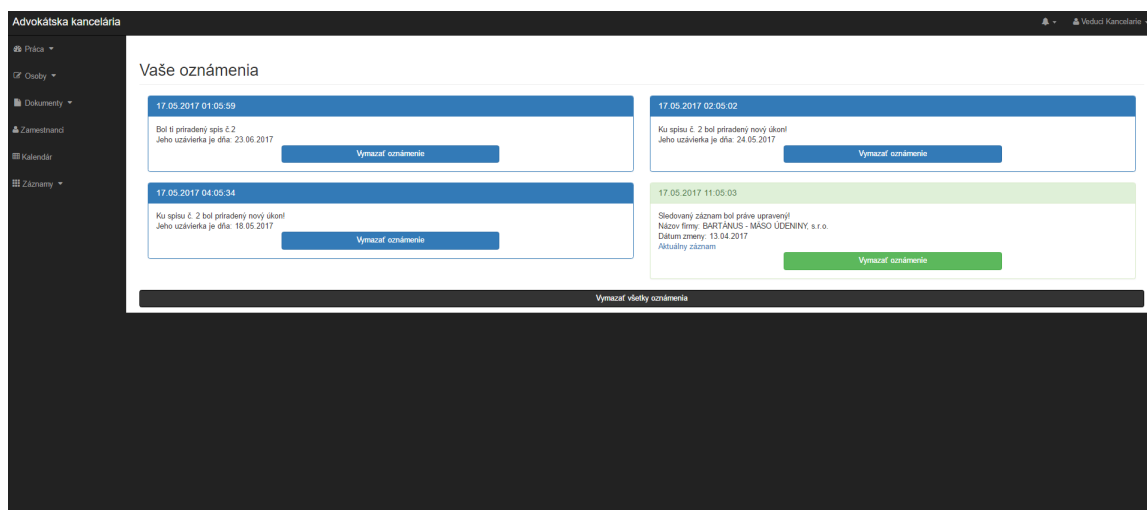
- `/cfg/` – Možné konfiguračné súbory.
- `/dbs/` – Testovacie databázy.
- `/law_office/` – Koreňový adresár Symfony aplikácie.
- `/pdf/` – Adresár obsahujúci technickú správu.
- `/tex/` – Adresár obsahujúci súbory potrebné na vygenerovanie \LaTeX dokumentácie.
- `Readme.txt` – Súbor obsahujúci dodatočné informácie.

Príloha B

Ukážka užívateľského rozhrania



Obr. B.1: Zobrazenie detailu úkonu.



Obr. B.2: Zobrazenie oznámení.