



Návrh a realizace webového portálu se zohledněním aktuálních trendů vývoje aplikací včetně podpory mobilních zařízení

Diplomová práce

Studijní program: N6209 – Systémové inženýrství a informatika

Studijní obor: 6209T021 – Manažerská informatika

Autor práce: **Bc. Jan Cabalka**

Vedoucí práce: Ing. Petr Weinlich, Ph.D.



ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Jan Cabalka**
Osobní číslo: **E14000386**
Studijní program: **N6209 Systémové inženýrství a informatika**
Studijní obor: **Manažerská informatika**
Název tématu: **Návrh a realizace webového portálu se zohledněním aktuálních trendů vývoje aplikací včetně podpory mobilních zařízení**
Zadávací katedra: **Katedra informatiky**

Z á s a d y p r o v y p r a c o v á n í :

1. Problematika technické stránky vzniku webových projektů
2. Analýza vybraného projektu
3. Tvorba moderní webové aplikace
4. Doplnění dalších funkcí prostřednictvím aplikací pro mobilní zařízení
5. Zhodnocení výsledků projektu

Rozsah grafických prací:

Rozsah pracovní zprávy: **65 normostran**

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

GASSTON, Peter. Moderní web. Brno: Computer Press, 2015.

ISBN 978-80-251-4345-2.

PILGRIM, Mark. Ponořme se do HTML5. Praha: CZ.NIC.

ISBN 978-80-905802-6-8.

ŘEZÁČ, Jan. Web ostrý jako břitva: návrh fungujícího webu pro webdesignery a zadavatele projektů. Jihlava: Baroque Partners, 2014.

ISBN 978-80-87923-01-6.

LACKO, Ľuboslav. Vývoj aplikací pro Android. Brno: Computer Press, 2015.

ISBN 9788025143476.

PETZOLD, Charles. Creating Mobile Apps with Xamarin. Forms. 2nd ed.

Redmond, WA: Microsoft Press, 2015. ISBN 978-0-7356-9723-2.

Elektronická databáze článku ProQuest (knihovna.tul.cz).

Vedoucí diplomové práce:

Ing. Petr Weinlich, Ph.D.

Katedra informatiky

Konzultant diplomové práce:

Ing. Kamil Vašák

iDomino a. s.

Datum zadání diplomové práce: **31. října 2015**

Termín odevzdání diplomové práce: **31. května 2017**



doc. Ing. Miroslav Žižka, Ph.D.
děkan



doc. Ing. Jan Skrbek, Dr.
vedoucí katedry

V Liberci dne 31. října 2015

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum:

Podpis:

Poděkování

Na tomto místě bych rád poděkoval Ing. Petru Weinlichovi, Ph.D. za odborné vedení diplomové práce, rady a cenné připomínky, dále celému týmu okolo portálu Reklamář.cz za vznik myšlenky a za pomoc s následnou realizací celého unikátního portálu. Velký dík patří také mé rodině a přítelkyni za podporu a povzbuzování po celou dobu mého studia.

Anotace

Cílem této diplomové práce je navrhnout za pomoci aktuálních trendů vývoje aplikací moderní webový portál, který bude optimalizován pro klasická PC i pro mobilní zařízení. V první části práce jsou charakterizovány současné aspekty technické stránky vzniku webových portálů, zejména HTML5 a CSS3. Práce se zabývá problematikou zabezpečení webových aplikací a pokročilejšími přístupy k tvorbě databází. V další části je provedena analýza vznikajícího projektu Reklamář.cz, v níž dochází k definování požadavků, jimiž se řídí vývoj webového portálu popsany v následující kapitole. V předposlední kapitole je analyzována problematika rozšíření funkcí webového portálu prostřednictvím aplikace pro mobilní zařízení, jejímiž specifiky se autor řídil při následném vývoji reálné aplikace. V závěru jsou provedeny výsledné ekonomické analýzy a zároveň shrnuty přínosy této práce.

Klíčová slova

Webový portál, Reklamář.cz, mobilní zařízení, HTML5, CSS3, zabezpečení webové aplikace, databáze

Annotation

Design and realization of a web portal with consideration of current trends in the applications development, including support for mobile devices

The aim of this diploma thesis is to design a modern web portal with usage of the current trends of application development that will be optimized for classical PC's and mobile devices. In the first part of this thesis are described the current technical aspects of web portal creation, especially HTML5 and CSS3. Thesis deals with web application security and advanced approaches for creating databases. The next section is an analysis of the emerging project Reklamář.cz, which is to define the requirements governing the development of the website, as described in the next chapter. In the penultimate chapter is analyzed problematic of web portal expansion by application for mobile devices, whose specific author uses in the subsequent development of real application. At the end there are formulated benefits of this thesis and the resulting economic analysis.

Keywords

Web portal, Reklamář.cz, mobile devices, HTML5, CSS3, web application security, database

Obsah

Seznam obrázků.....	12
Seznam zkratk.....	13
Úvod.....	14
Literární řešerše.....	16
1 Problematika technické stránky vzniku webových projektů.....	17
1.1 Definice webového portálu.....	17
1.2 Rozdělení dle funkčnosti.....	18
1.3 Dostupné technologie.....	18
1.3.1 Vrstva obsahu.....	19
1.3.2 Prezentační vrstva.....	21
1.3.3 Interakční vrstva.....	22
1.3.4 Kompletní řešení.....	22
1.4 Moderní web.....	23
1.4.1 HTML5, CSS3.....	24
1.4.2 Flash – postupný zánik.....	27
1.4.3 Různé platformy.....	29
1.4.4 Javascript, JQuery.....	30
1.4.5 Responsivní weby.....	31
1.4.6 Očekávání uživatelů.....	33
1.5 Zabezpečení webového portálu.....	35
1.5.1 Práce s hesly.....	35
1.5.2 Komunikace s databází.....	37
1.5.3 HTTPS.....	39
1.5.4 Uživatelská práva u databází.....	39
1.5.5 Cíle pro moderní web.....	40

1.6	Databáze.....	40
1.6.1	Procedury a vlastní funkce	41
1.6.2	Pohledy	41
1.6.3	Transakce.....	42
1.7	Plán vývoje	42
1.7.1	Plánování cílů	43
1.7.2	Plánování požadavků.....	44
1.7.3	Návrh zvoleného řešení	45
1.7.4	Implementace do testovacího prostředí	46
1.7.5	Celkové testování	46
1.7.6	Zavádění systému	47
1.7.7	Zkušební období	47
1.7.8	Rutinní provoz a údržba	48
1.7.9	Zásadní přehodnocení funkčnosti	48
2	Analýza vybraného projektu.....	49
2.1	Základní koncepce (plánování cílů).....	49
2.1.1	Důvod vzniku portálu	50
2.1.2	Současný stav	50
2.1.3	Plán realizace.....	51
2.1.4	Odhady časových a finančních náročností	52
2.2	Požadovaná funkčnost (plánování požadavků).....	52
2.3	Diagramy	53
2.3.1	Use Case – diagram uživatelských rolí	54
2.3.2	Diagram aktivit – proces zadávání dokumentů a reklamací.....	56
2.3.3	Diagram aktivit – proces obnovy hesla	58
2.4	Specifika projektu	60

2.5	Výběr technologie	61
2.6	Podklady pro výběr vhodného webhostingu	62
3	Tvorba moderní webové aplikace	63
3.1	Databázový model (fáze návrhu)	63
3.2	Vývoj moderního webového portálu (fáze návrhu)	69
3.3	Reálná implementace do testovacího prostředí	72
3.4	Celkové testování	72
3.5	Přípravy k provozu (fáze zavádění systému)	73
3.6	Zkušební období	73
3.7	Rutinní provoz, údržba a případné optimalizace	74
3.8	Výsledný portál	74
3.8.1	Hlavní stránky	74
3.8.2	Klientská část	75
3.8.3	Administrátorská část	76
3.8.4	Automatizovaný skript	77
4	Doplnění dalších funkcí prostřednictvím aplikací pro mobilní zařízení	78
4.1	Analýza předpokladů	79
4.2	Výběr vhodné technologie	81
4.3	Návrh aplikace dle zvolené technologie	82
4.4	Vývoj	82
4.5	Testování	82
4.6	Publikování aplikací	83
4.7	Zhodnocení vývoje mobilní aplikace	84
5	Zhodnocení výsledků projektu	85
5.1	Technické přínosy	85
5.2	Ekonomická a časová analýza	85

5.3	Budoucnost projektu	86
	Závěr	87
	Seznam použité literatury	89
	Seznam příloh	93
	Příloha A – Úvodní stránka portálu	94
	Příloha B – Zobrazení úvodní stránky při změně horizontálního rozlišení	96
	Příloha C – Záznamy z testování	97
	Příloha D – Vybrané stránky portálu	99

Seznam obrázků

<i>Obrázek 1: Stejný obsah na různých zařízeních</i>	32
<i>Obrázek 2: Příklad SQL Injection</i>	38
<i>Obrázek 3: Logo portálu Reklamář.cz</i>	49
<i>Obrázek 4: Use case diagram portálu Reklamář.cz</i>	54
<i>Obrázek 5: Proces zadávání dokladů a reklamací</i>	57
<i>Obrázek 6: Proces obnovy hesla</i>	59
<i>Obrázek 7: Databáze portálu Reklamář.cz</i>	64
<i>Obrázek 8: Procedura sp_doklady_polozky</i>	68
<i>Obrázek 9: Graf rozložení časů při používání mobilních zařízení uživateli iOS a Android</i>	79
<i>Obrázek 10: Časová analýza vývoje portálu Reklamář.cz</i>	86

Seznam zkratek

API	Application Programming Interface
ARES	Administrativní registr ekonomických subjektů
CMS	Content Management System
CSS	Cascading Style Sheet
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secured
MSSQL	Microsoft SQL Server
PDO	PHP Data Objects
SDLC	System Development Life Cycle
SEO	Search Engine Optimization
SQL	Structured Query Language
UA	Účet pro administraci webového portálu
UW	Účet pro webový portál
UX	User Experience
W3C	World Wide Web Consortium
WYSIWYG	What You See Is What You Get
XHTML	Extensible Hypertext Markup Language
XML	Extensible Markup Language

Úvod

Webové portály jsou v moderní době nepostradatelnou součástí každodenního života velké části lidské populace. Jejich význam stále narůstá nejen v oblasti informatiky, ale také v rámci ekonomie a podnikání. Dnes je nepsaným standardem, že se prakticky každý ekonomicky činný subjekt prezentuje prostřednictvím webových stránek. Pokud se podnikatelské aktivity zaměřují na prostředí internetu, je velice důležité podpořit je kvalitně vytvořeným webovým portálem. Cílem této diplomové práce je vyvinout moderní webový portál, který bude reflektovat aktuální vývojové trendy v rámci tvorby aplikací a současně bude optimalizován pro PC a pro mobilní zařízení.

Teoretická část práce se zaměřuje na problematiku technické stránky vzniku webových projektů. Nejprve se autor zabývá otázkou, jaké technologie jsou momentálně pro vývoj webového portálu k dispozici. V kapitole Moderní web je definováno kritérium použití aktuálních trendů, dále se tento oddíl zaměřuje na nezbytné aspekty pro podporu mobilních zařízení. Pro vývoj webu je neméně důležitou částí webová bezpečnost, jíž je věnována následující kapitola. Webová bezpečnost je jedním z klíčových témat v rámci problematiky vzniku webových portálů, neboť kvůli vzrůstajícím počtům útoků vznikají stále nové možnosti, jak tyto portály zabezpečit. V závěru teoretické části se autor zaměřuje na pokročilejší přístupy k tvorbě databází a definuje plán vývoje pro jasné ohraničení jednotlivých částí v rámci životního cyklu celého projektu.

Praktická část práce se zabývá analýzou projektu webového portálu Reklamář.cz, který vznikl souběžně s touto prací. Analýza se zaměřuje nejprve na základní koncepci celého portálu. Z této koncepce poté vychází při plánování požadované funkčnosti a plánovaných specifik projektu. Následně jsou vybrány potřebné technologie pro realizaci a společně s webhostingem, na němž je portál provozován.

Na základě provedených analýz je v následující kapitole popsána fáze realizace vývoje a nasazení moderního webového portálu. V rámci tohoto vývoje jsou zohledněny aktuální trendy definované v teoretické části včetně dnes nezbytné podpory mobilních zařízení. Vývoj je zahájen tvorbou databázové struktury a následně samotné webové aplikace, řídí se zvolenou metodikou a jeho cílem je reálné nasazení portálu do produkčního prostředí.

Předposlední část této práce je věnována otázce rozšíření funkčnosti webového portálu pomocí aplikací pro mobilní zařízení. Dochází zde k analýze dostupných technologií a dalších předpokladů pro vznik aplikací. Nad těmito technologiemi je následně zahájen vývoj mobilní aplikace, jejíž podstatou je podpora nejrozšířenějších operačních systémů pro mobilní zařízení.

V závěrečné kapitole této práce autor definuje přínosy a formuluje výsledky, jejichž součástí jsou ekonomické a časové analýzy.

Literární rešerše

Protože se technická stránka vývoje webových portálů neustále vyvíjí, je dostupné široké spektrum odborné literatury, odborných článků a webů, ze kterých bylo možné čerpat cenné informace a rady pro tuto práci.

Stěžejní informace týkající se dané problematiky byly obsaženy zejména v publikaci Petera Gasstona „*Moderní web*“, kde autor popisuje techniky a nástroje pro tvorbu webových stránek s přihlédnutím k funkčnosti v rámci různých (i mobilních) zařízeních. Souběžně jsou zde definovány přístupy k tvorbě aplikačních rozhraní a techniky pro testování a ladění webových stránek. [1]

Pro získání lepšího povědomí o HTML5 autorovi posloužila kniha Marka Pilgrima „*Ponořme se do HTML5*“. Tato kniha se věnuje standardizovaným komponentám značkovacího jazyka a definuje nové přístupy k tvorbě webových stránek. [2]

Problematikou návrhu a tvorby webových stránek se zabývá Jan Řezáč, jeden z nejpopulárnějších českých konzultantů pro webové stránky, v publikaci „*Web ostrý jako břitva: návrh fungujícího webu pro webdesignery a zadavatele projektů*“. Autor v knize prochází celý proces vývoje od poptávky až po nasazení webu. Zároveň poskytuje cenné rady, jak se vyhnout zásadním chybám při návrhu koncepce webových portálů. [3]

Téma moderních webových portálů je velmi obsáhle zpracováno i v placené databázi ProQuest, do které mají studenti Technické Univerzity v Liberci přístup. Například článek „*Modern Web Application Systems: What It Really Means*“, jehož autorem je Rossane Doyle, pojednává o přínosech použití moderních webových portálů v rámci podnikání. [4]

Na oblast vytváření mobilní aplikace se zaměřuje kniha Luboslava Lacka „*Vývoj aplikací pro Android*“ zabývající se zejména specifiky webových aplikací. Součástí knihy jsou i popisy aplikační logiky pro napojení na externí systémy. [5]

V neposlední řadě autor využil informací obsažených v knize Charlese Petzolda „*Creating Mobile Apps with Xamarin.Forms*“, jež se zabývá kompletní specifikací nástrojů a realizovatelných postupů při vývoji mobilních aplikací v rámci vývojového prostředí Xamarin. [6]

1 Problematika technické stránky vzniku webových projektů

1.1 Definice webového portálu

Co je webový portál? Webový portál je sám o sobě široký pojem zastřešující spoustu dílčích částí, které dohromady tvoří funkční celek. Základní stavební jednotkou webového portálu je webová stránka, která obsahuje informace z různých zdrojů prezentované v ucelené formě.

Webový portál možné rozdělit na tři základní typy dle konkrétní funkčnosti [3]:

1. **Webová prezentace** je druh portálu, který slouží k prezentování produktu / firmy / člověka. Uživatel tyto portály navštěvuje s cílem získat větší povědomí o prezentovaném produktu a nalezení všech potřebných informací. Dnes se již stává nepsaným standardem, že má každá firma své vlastní webové stránky.
2. **E-shop** navazuje na princip webové prezentace, který rozšiřuje funkci prezentovat daný obsah o možnost elektronického obchodování s produkty. Uživatel od tohoto typu portálu očekává rychlost, snadnost a přehlednost obchodu. Vlastník naopak očekává, že portál bude pro případné klienty zajímavý a bude sloužit k jejich ovlivnění směrem ke koupi a tím i zisku.
3. **Webová aplikace** je jinými slovy webový nástroj pro řešení konkrétního problému. Úspěšnou webovou aplikaci lze poznat tak, že uživatel si na ni velice rychle vytvoří návyk; aplikace se tak stává běžnou součástí jeho života. Hlavním cílem zde již není prezentace či prodej (i když to mohou být případné vedlejší cíle), ale úspěšně řešení zadané problematiky.

Pro možnost snadné editace zobrazeného obsahu mívají některé portály i redakční systém (CMS - Content Management System), což je nástroj pro správu obsahu.

Při zahájení tvorby vlastního webového portálu je zapotřebí věnovat nemalou pozornost návrhu optimálního řešení – i drobná chyba v tomto počátečním návrhu totiž může v důsledku přivodit množství práce a problémů navíc. Proto by na počátku tvorby

měly existovat analýzy. Ty se mohou zabývat například konkurenčním prostředím, plánům budoucího rozvoje či celé požadované funkčnosti.

1.2 Rozdělení dle funkčnosti

Pro to, abychom mohli hlouběji nahlédnout do problematiky technické stránky vzniku, je nutné rozlišit webové stránky statické a webové stránky dynamické.

Statické webové stránky jsou jednoduchými, navzájem prolinkovanými soubory na síťovém uložišti, jejichž obsah je zpravidla v čase neměnný. Případná úprava dat znamená pro správce přímou úpravu zdrojových kódů. Tento druh stránek se využívá zejména pro webové prezentace, u nichž zadavatel neplánuje informace aktualizovat příliš často. Mezi jejich hlavní výhody patří vysoká rychlost, snadný vývoj a nízké náklady na provoz. Nevýhodou je, že obsah nemusí být vždy aktuální a je zde nutný zkušený uživatel, který bude zajišťovat úpravy obsahu.

Dynamické webové stránky jsou stránky zpracovávané serverem tak, aby uživatelé zobrazily různá data v závislosti na zadaných vstupech. Tyto stránky zpravidla využívají pokročilejší nástroje pro správu obsahu a databáze. U webových stránek s redakčním systémem je možné velmi rychle upravovat obsah a měnit případně i jeho strukturu. Součástí dynamických webů mohou být i oddělené klientské sekce či integrované webové aplikace. Nevýhodou zde jsou vyšší náklady spojené s vývojem a provozem (potřeba provozování webového serveru podporujícího jazyk, ve kterém je stránka napsána, a zpravidla i zvolené databáze). [7]

1.3 Dostupné technologie

Dnes je na trhu k dispozici široká škála dostupných technologií, které se liší svým pojetím i zaměřením. Pro to, abychom dokázali správně rozlišit jednotlivé druhy technologií, nám poslouží všeobecně uznávané rozdělení na 3 základní vrstvy:

- Vrstva obsahu – zjednodušeně řečeno to, co uživatel na webové stránce uvidí
- Prezentační vrstva – určuje, jak se obsah zobrazí uživateli
- Interakční vrstva – řeší, jak bude uživatel schopen ovládat obsah

1.3.1 Vrstva obsahu

V této vrstvě jsou definovány jednotlivé texty a multimediální prvky obsahu (obrázky, zvuk, video). U statických webů je součástí této vrstvy pouze datové uložení, na kterém jsou stránky provozovány.

Naopak u dynamických webů jsou již potřeba pokročilejší technologie pro správu dat – databázové systémy. Těchto systémů je dnes široká škála. Jednotlivé databázové systémy se pak mohou od sebe lišit například odlišným zaměřením, způsobem uchování dat či způsobem dotazování na data. V následujícím rozdělení se autor zaměří na tři systémy, které jsou dle webu db-engines, jenž pravidelně monitoruje podíl používaných databází v celosvětovém měřítku, na trhu nejrozšířenější. Všechny tyto databáze vycházejí ze základů jazyka SQL (Structured Query Language – strukturovaný dotazovací jazyk), který volně implementují a rozšiřují (některé příkazy jsou pro každý systém odlišné – např. příkaz LIMIT pro získání konkrétního počtu řádků z databáze). [3]

Nejrozšířenějšími systémy jsou v době psaní této práce:

1. **Oracle database** – placené databázové řešení firmy Oracle specifické tím, že je podporováno širokou škálou programovacích jazyků. Výhodou je též fakt, že je možné provozovat jej na velkém spektru různých platform (Windows, UNIX, Linux, Mac OS, atd.). Své příznivce si získává díky svým vysokým výkonům a velmi dobré škálovatelnosti.
2. **MySQL** – open-source řešení vytvořené firmou MySQL AB, nyní spadající pod firmu Sun Microsystems. Díky své cenové politice a jednoduchosti si získalo velkou oblibu malých a středních webových projektů. Téměř každý webhosting nabízí alespoň jeden balíček, jehož je MySQL součástí.
3. **Microsoft SQL Server (MSSQL)** – placené řešení používané především ve firemní sféře. Díky integrovaným nástrojům pro správu databází a pro tvorbu pokročilých analýz je velmi populární ve firmách používajících serverové řešení od firmy Microsoft. Nespornými klady MSSQL jsou možnosti jednak rychlého zpracování velkého množství dat a jednak napojení na vývojářské nástroje Microsoftu.

Nutností pro komunikaci s databází a pro práci s daty je využití programovacích jazyků. Mezi nejrozšířenější programovací jazyky patří:

- a) **PHP** – v současné době nejrozšířenější objektově orientovaný serverový programovací jazyk (dle statistiky W3Techs má v době psaní této diplomové práce 82,2% podíl na trhu programovacích jazyků pro webové stránky). Mezi jeho hlavní výhody patří licence Open-Source, široká online komunita pro podporu vývojářů a také to, že je možné je provozovat na různých platformách. Další nespornou výhodou je také to, že může být integrován přímo do HTML syntaxe webu. Výhodou PHP také je, že je nad ním postaveno velké množství tzv. frameworků – předpřipravených knihoven vytvořených pro usnadnění práce vývojářům tím, že mohou využívat již hotových komponent. Nejznámějšími frameworky jsou například Laravel, Nette či Zend. PHP používá například sociální síť Facebook či redakční systém WordPress. [4]
- b) **Jazyky platformy .NET** – platforma firmy Microsoft využívající prostředí Windows složená ze souboru technologií pro tvorbu nejen webových portálů, ale i ostatních aplikací. Tato platforma je velmi rozšířena ve firemní sféře, kde jsou díky ní vytvářeny velké vnitropodnikové aplikace a portály. Nejrozšířenějšími programovacími jazyky, které tato platforma využívá, jsou:
 - a. *C#* - objektově orientovaný jazyk založený na C++ a Javě;
 - b. *Visual Basic .NET* – objektově orientovaná verze jazyka Visual Basic, který umožňuje pomocí jazyka Basic vytvářet aplikace s grafickým rozhraním.Typickými portály, které využívají platformy .NET, jsou například weby společnosti Microsoft (a také webový vyhledávač Bing).
- c) **Java** – objektově orientovaný jazyk, dnes využívaný převážně pro rozsáhlé webové portály s předpokládanou vysokou návštěvností. Hlavními výhodami jsou vysoká výkonnost, multiplatformnost, velké množství předpřipravených a otestovaných knihoven. Nejznámějšími webovými portály využívajícími Javu jsou profesní sociální síť LinkedIn či aukční portál Ebay. Na Javě je také postaven operační systém Android od firmy Google.
- d) **Python** – pokročilejší jazyk zaměřující se především na čitelnost kódu – pomocí méně řádků kódu je možné napsat více funkcionalit než v ostatních jazycích. Navržený pro velké portály. Je snadno implementovatelný do stávajících systémů.

Nejznámějšími příklady využití tohoto jazyka jsou zejména Instagram (aplikace pro sdílení fotografií) a Pinterest (sociální „náštnková“ síť).

- e) **Ruby** – alternativa k jazyku Python. Vychází ze stejné filosofie – tj. že produktivní zdrojový kód by měl být úsporný a snadný pro tvorbu. Na tomto programovacím jazyku je postavený Framework Ruby on Rails (někdy označovaný jen jako Rails), pomocí kterého jsou vytvořeny například Twitter (sociální síť pro sdílení krátkých zpráv) a digitální online knihovna Scribd.

Programovacích jazyků existuje samozřejmě na trhu více než zmíněných pět. Tyto další jazyky neustále vznikají a zanikají, různě se slučují či transformují do podoby různých frameworků.

1.3.2 Prezentační vrstva

Pokud máme připravený obsah, který chceme publikovat, měli bychom mu dát jasnou strukturu a určit, jak bude na konkrétní stránce vypadat. K tomu jsou využívány značkovací jazyky. V současné době jsou nejpoužívanějšími jazyky:

- **HTML (HyperText Markup Language)** – jazyk využívaný pro tvorbu základní struktury webové stránky určující jak zobrazit data. Využívá pevně definovaný formát jednotlivých značek (tzv. tagů), které jsou dle standardu interpretovány webovou stránkou (například tag „*strong*“ pro tučný text či „*table*“ pro zobrazení tabulky).
- **XML (eXtensible Markup Language)** – jazyk pro tvorbu strojově zpracovatelných stránek určující jak data popsat. Tento jazyk nemá předem definované tagy, které by se v případě využití HTML měly dle různých pravidel interpretovat. XML samo o sobě definuje syntax a soubor pravidel, podle kterých uživatel ohraničí jednotlivá data vlastními tagy. Je tak možné sdílet obsah mezi jednotlivými weby či mobilními aplikacemi.
- **XHTML (eXtensible HTML)** je HTML rozšířené o pravidla XML. Protože jazyk HTML nebyl dlouho od uvedení verze 4.01 rozvíjen a na trhu bylo zapotřebí vytvořit standard, který by opravil v té době časté nedostatky HTML (docházelo k různým interpretacím jednotlivých tagů mezi tvůrci prohlížečů; vznikaly individuální tagy a jádra prohlížečů tak nebezpečně rostla; navíc klasické HTML bylo benevolentní k chybám a špatně formátované tagy interpretovalo různými

způsoby). V současné době je však postupně z trhu vytlačováno pomocí HTML verze 5. [2]

Máme-li definovanou strukturu dat, je třeba zabývat se popisem vizuálního zobrazení jednotlivých elementů webových stránek. K tomu slouží jazyk **CSS (Cascading Style Sheet)**, který je aktuálně ve verzi 3. Pomocí tohoto jazyka lze každému prvku webové stránky určit, jakou má mít pozici, jaký má mít vzhled, jak má být formátovaný text, atd. Silnou stránkou CSS3 je možnost definovat různé rozložení prvků pro různé typy zařízení. Více viz v kapitole Responsivní weby. [1]

1.3.3 Interakční vrstva

Ve fázi, kdy má uživatel zobrazená a naformátovaná data, jsou volitelnou částí webové stránky komponenty, pomocí nichž lze webové stránky ovládat. Tyto komponenty se nezpracovávají na straně serveru (jako u prezenční vrstvy), ale u klienta v prohlížeči. Nejrozšířenějším nástrojem je objektově orientovaný skriptovací jazyk **JavaScript**. Tento jazyk původně začala vyvíjet firma Netscape s cílem přidat dynamické a interaktivní prvky do webových stránek. Dnes jsou součástí JavaScriptu funkce pro obsluhu široké škály událostí na webové stránce (například funkce `onClick` pro zpracování kódu při kliknutí na prvek uvnitř webové stránky či `onSubmit` pro zpracování kódu při odeslání formuláře).

Na JavaScriptu je postavena široká škála knihoven, pomocí kterých můžeme do webové stránky snadno přidávat různé komponenty (nástroje pro zpracování tabulek, grafů, souborů, atd.)

Dalším nástrojem prezentační vrstvy po JavaScriptu je **Flash**, který však dnes přestává být doporučován, v některých případech i podporován (více viz kapitola Flash).

1.3.4 Kompletní řešení

Při návrhu webového portálu je zapotřebí zvážit, zdali se vydat cestou tvorby vlastního řešení nebo využít již hotových nástrojů, které jsou nad technologiemi definovanými v předchozích kapitolách vytvořeny.

Nejrozšířenější jsou v době psaní této diplomové práce nástroje:

- a) **WordPress** – CMS systém s open-source licencí postavený na PHP a MySQL. Díky své cenové politice (systém je zdarma, platí se za grafické šablony a moduly

s rozšiřujícími funkcemi) a minimálním nárokům na zprovoznění se stal mezi komunitou webových vývojářů velmi populárním nástrojem pro tvorbu malých a středních webových portálů. Populární je také u autorů píšících webové blogy.

- b) **Prestashop** – nástroj pro tvorbu elektronických obchodů s open-source licencí postavený také na PHP a MySQL. Pro grafické šablony využívá Smarty (nástroj oddělující aplikační logiku webové stránky od prezentační). Stejně jako u WordPressu je jeho cenová politika postavená na placených šablonách a rozšiřujících modulech.
- c) **SharePoint** – placené řešení firmy Microsoft pro tvorbu nejen webových portálů organizací, ale i aplikací pro správu dokumentů, uchovávání dat či plánování datových toků. Tato sada nástrojů je vytvořena v prostředí ASP.NET a využívá MSSQL databázi. Díky svým možnostem vytváření široké škály aplikací je značně rozšířen u firem, které již používají některé z řešení Microsoftu pro firmy.

1.4 Moderní web

V rámci problematiky moderního webu je třeba zodpovědět několik zásadních otázek: Co je moderní web? Co všechno jeho tvorba obnáší? Jakým směrem by se měl každý zájemce o moderní web vydat? Jaké jsou na trhu dostupné technologie a které trendy je zapotřebí zohlednit? Těmito otázkami se bude autor zabírat v rámci následujících kapitol.

Websterův slovník definuje pojem „moderní“ jako „*vztah k současnosti či nedávné minulosti: odehrávající se, existující, či vyvíjené v čase blízkém či současném*“. Pojem moderní v sousloví moderní informační systém je ve slovníku definován jako „*systém založený na nejnovějších informacích, metodách nebo technologiích či systém takovýchto informací využívající*“. [10]

Gaston uvádí, že moderní přístup k tvorbě webových stránek řeší tvorbu stránek fungujících napříč různými zařízeními a zohledňuje plné využití zobrazení u každého zařízení s přihlédnutím k jeho rozměrům a schopnostem. [1]

Penny McIntire tuto charakteristiku moderního webu doplňuje o jeho další funkci, kterou je rozvoj komunikace s návštěvníky webu. Při této komunikaci by měl být vizuální styl stránek poutavý, ale ne zbytečně naléhavý.[8]

Dle Jeffreyho Vocella, Product Marketing Managera ve společnosti HubSpot, se každý rok objevují na trhu nové prvky, styly a technologie ve tvorbě moderního webu, přičemž některé z nich mohou být značným přínosem. Tento přínos není automatický, je proto na tvůrci, aby zvážil, které prvky využít a které ne, aby výsledný efekt měl pro vlastníka webu co největší užitek.[9]

Aby se dal web označit za moderní, měl by:

- využívat nejnovější (ale pokud možno odladěné) technologie,
- splňovat požadavky na bezpečnost dat,
- být zobrazitelný napříč různými platformami včetně mobilních zařízení.

Jak dosáhnout splnění těchto kritérií bude vysvětleno v následujících kapitolách.

1.4.1 HTML5, CSS3

HTML5 vznikl jako nástupce standardu HTML 4.01 (standardu vydaného v roce 1999) a alternativa XHTML2. Vývoj byl zahájen roku 2007 a finální specifikace byla schválena až v roce 2014. Tento standard přináší několik nových elementů pro tvorbu webových stránek, některé zastaralé či nepoužívané elementy naopak ruší. Jedním z cílů HTML5 je také standardizace různých přístupů, které vývojáři při vývoji webových stránek používaly pro úpravu chování standardních elementů. [2]

Verze 5 jazyka HTML přináší tyto změny:

- **Nové struktury webových stránek** – nové tagy zvyšují čitelnost zdrojových kódů určováním jejich účelu na stránce. Původní struktura webové stránky byla složena z mnoha tagů „div“, moderní přístup je nahrazuje či doplňuje tagy „header“, „nav“, „section“ apod.
- **Zjednodušení hlaviček** – mnohdy složité a na interpretaci náročné hlavičky jsou nyní zjednodušeny do prostého tvaru. Není již nutné složitě zadávat dle jakého konkrétního standardu se má stránka zobrazit a jaká je používaná znaková sada – vše je možné vyřešit zkráceným zápisem.
- **Rozšíření formulářů** – u předchozí verze standardu HTML bylo možné určit jen pár základních typů formulářových vstupů a jejich vyplnění či validitu následně ověřovat pomocí dopsaných JavaScriptových funkcí. Nový standard přidává

rozšiřující vlastnosti, které umožňují určit, zdali formulářový prvek má od uživatele vyžadovat zadání data, adresy URL, emailu, čísla či vybírat ze zvoleného rozsahu. Tyto údaje jsou ihned při vkládání validovány. Pokud je využit i nový atribut „*required*“ a formuláře není možné odeslat ke zpracování, dokud toto pole nebude řádně vyplněno.

- **Podpora audio a video souborů** – pomocí nových tagů je možné do stránky integrovat multimediální obsah, přičemž tento obsah bude přehrán pomocí přehrávače integrovaného ve webovém prohlížeči. Tím je možné nahradit nutnost integrovat do webů externí knihovny s přehrávači, které jsou z větší části založeny na Flashi (více viz následující kapitola).
- **Microdata** – doplňují atributy ke stávajícím tagům pro zlepšení strojové čitelnosti webových stránek. Cílem je zlepšit interpretaci dat uvnitř webové stránky pro automatické roboty prohledávající a indexující web. Na základě správné interpretace může být stránka lépe zobrazena ve webovém prohlížeči či ve webovém vyhledávači. Reálným příkladem použití může být například profil osobnosti, u jehož jednotlivých tagů je dle standardu určeno, že se jedná o jméno osobnosti, její foto či například odkaz na firemní stránky.
- **Canvas** – nástroj pro kreslení uvnitř webového prohlížeče. Pomocí tohoto „plátna“ a JavaScriptu je možné na webové stránce vykreslovat a manipulovat s grafikou. Tento nástroj dokáže dynamicky tvořit různé grafy a animace, příkladem využití může být zobrazení vlastní mapy.
- **Fulscreen a Geolocation API** – pomocí rozhraní Fulscreen je možné přepnout dokumenty a obrázky do rozšířeného zobrazení přes celou obrazovku. Geolocation API naopak umožňuje stránce vznést dotaz na globální pozici uživatele.
- **Další funkcionalita** – například integrovaná podpora Drag & drop (způsob kopírování souboru při přetažení) či API pro komunikaci mezi webovými stránkami Web messaging.

V průběhu druhé poloviny roku 2016 je naplánováno schválení verze HTML5.1. Ta by měla mimo jiné přinést nové tagy či podporu tvorby vlastních dialogových oken. Následně bude záležet na době, v jaké stihnou vývojáři webových prohlížečů tento standard implementovat do vykreslovacích jader, a také na uživateli, kteří si budou své stávající

webové prohlížeče aktualizovat. Jakmile budou na trhu převládat aktualizované prohlížeče, bude možné začít nové funkce využívat vývojáři při tvorbě moderních webových stránek.

HTML5 samo o sobě přináší spoustu nových funkcí, nicméně moderní web potřebuje i moderní vzhled a k tomu je nutné využít CSS, nejlépe jeho nejaktuálnější verzi CSS3.

CSS3 vzniklo jako nástupce verze CSS2.1, která byla publikována roku 2011. Oproti svým předchozím verzím se aktuální verze odlišuje tím, že její součásti vznikají jako samostatné moduly, které jsou postupně vyvíjeny a publikovány. Nyní již postupně vzniká též návrh specifikace nástupce – CSS4. Následující soupis vlastností pojednává o uznaných a doporučených modulech CSS3 [13].

Důležitými prvky CSS3 jsou:

- **Nové selektory** – nově vzniklé selektory jsou užitečné pro přesnější výběr prvku na stránce určeného ke stylování. Umožňují například aplikovat odlišné styly na prázdné elementy nebo na povolené/ zakázané / zaškrtnuté formulářové prvky; či pracovat s pokročilejšími hierarchiemi v kódu, tzn. určit vybraného (prvního, posledního, jediného či n-tého) potomka svého rodiče a prvního, posledního, či jediného elementu dle typu.
- **Border radius** – tento parametr umožňuje prvkům na webové stránce mít kulaté rohy. Do doby před vznikem této vlastnosti bylo nutné kulaté rohy vytvářet pomocí dodatečných grafických souborů a jejich následného složitého umístování do stránek pomocí pozicování.
- **Box-sizing** – vlastnost upravující zobrazení prvků na stránce pomocí takzvaného „box modelu“, který definuje každý prvek jako objekt skládající se z obsahu, vnitřního okraje (padding) a vnějšího okraje (border). Při použití této vlastnosti lze prohlížeči říci, že požadovaný finální rozměr prvku stránky má být určen buď podle obsahu, podle vnitřního kraje či podle vnějšího kraje. Tato vlastnost je výhodná při tvorbě prvků, jejichž rozměry jsou dány v procentech (například v případě CSS 2.1 při použití 2 tagů „div“ na stránce zobrazených vedle sebe a majících velikost 50% by případné přidání 2px okraje znamenalo, že by se nemohly vejít na stránku – $50\% + 50\% + 2*2px + 2*2px > 100\%$)

- **Media-queries** (v české lokalizaci Dotazy na média) - vlastnost, kterou W3C (World Wide Web Consortium) uvádí jako doporučený standard od června 2012. V této době tak postupně začalo její rozšíření do webových prohlížečů s tím, že aktuálně ji podporují všechny prohlížeče jak pro mobily, tak i pro ostatní zařízení (včetně Internet Exploreru od verze 9). Media queries umožňují vývojáři zjistit, jaké možnosti zobrazení má zařízení přistupující na stránku. Podle toho lze následně určit, v jakém seskupení a zobrazení budou prvky stránky v prohlížeči interpretovány.
- **Transformace a škálování** – jednotlivé prvky je možné v rámci stránky různě upravovat – měnit jim velikost, pohybovat s nimi, otáčet je v prostoru či je různě škálovat.
- **Podpora stínů u textů a blokových prvků** – Dříve bylo nutné mít všechny stíny u blokových prvků řešené pomocí externí grafiky a jejího pozicování. Dnes je to možné řešit pouze za pomoci jedné vlastnosti. U textů byl tento problém ještě složitější. Chtěl-li mít autor na stránce text se stínem, musel tento text nejprve vytvořit v grafickém editoru jako obrázek. Tato praxe výrazně omezovala možnosti práce s textem a komplikovanější byly i případné změny obsahu.

Vývoj modulů CSS3 stále pokračuje. V nejbližší době bychom se měli dočkat například uznání modulu pro určení podmínek v CSS (Conditional Rules Module). Po odsouhlasení modulu bude vydáno doporučení k používání. Od této chvíle bude záležet pouze na tom, za jak dlouho začnou výrobci webových prohlížečů nové moduly podporovat a implementovat.

1.4.2 Flash – postupný zánik

Flash je technologie původně vyvinutá firmou Macromedia v roce 1996. Dnes je jejím vlastníkem firma Adobe. Tato technologie původně vznikla jako nástroj pro tvorbu složitějších animací, poté se postupně stala nepsaným standardem pro tvorbu interaktivního obsahu na webu. Díla vytvořená pomocí Flash je možné distribuovat i jako offline spustitelné soubory. [14]

Flash nebyl nikdy integrován do webových prohlížečů, takže pro jeho spuštění a chod je zapotřebí je nutné doinstalovat komponentu pro jeho zobrazení.

Na počátku 21. století došlo k obrovské expanzi Flashe na webu. S jeho pomocí totiž bylo možné vytvářet:

- Animované interaktivní bannery (reklamní proužky) či interaktivní schémata a prezentace.
- Kompletní webové stránky (obvykle menšího rozsahu, tzv. Microsite).
- Přehrávače hudby a videa (aktuální verze podporuje i standard H.264 pro kompresi videa).
- Hry (často jednoduššího ražení) umístěné na webu. Pomocí Flashe bylo možné mimo jiné rychle vytvářet i hry na podporu marketingu.
- Aplikace – pomocí odladěného programovacího jazyka bylo možné vytvářet aplikace na míru požadavkům klientů.

Navzdory všem vyjmenovaných možnostem využití je dnes Flash považován za zastaralou technologii. Moderní web (a dokonce i samo Adobe) postupně směřuje k pozvolnému nahrazování Flashe různými alternativami (zejména HTML5 a komponentami postavenými na Javasriptu).

Přestože má Flash své výhody, v průběhu posledních let k odhalení jeho výrazných slabin. Těmi jsou:

- **Hardwarové nároky** – oproti současným technologiím je Flash výrazně náročnější na výkon zařízení. Při zobrazování složitějších prvků Flash odkládá množství dat do dočasné paměti. To může být problémem při otevření více stránek s Flashovým obsahem.
- **Na mobilních zařízeních není podporován** – tento problém se týká hlavně zařízení pracujících s mobilními operačními systémy Google Android a Apple iOS. V průběhu vývoje byly ze strany Adobe realizovány pokusy o implementaci, avšak z důvodů malého výkonu mobilních zařízení a náročné implementaci nedošlo nikdy k jeho plnému nasazení.
- **Bezpečnostní problémy** – Flash bohužel obsahuje spoustu nebezpečných chyb. V posledních letech byly odhaleny chyby, které umožňovaly útočníkům potají do počítačů instalovat škodlivý software či dokonce vzdáleně celý počítač ovládnout.

Některé z těchto chyb byly opraveny, nicméně stále existují způsoby jejich zneužití (to je problém zejména neaktualizovaných zařízení).

V době psaní této diplomové práce již několikrát došlo k blokování celé funkčnosti Flashe ve webových prohlížečích (Chrome a Firefox) právě kvůli bezpečnostním problémům. Navíc stále roste podíl mobilních zařízení připojených na internet. Velké internetové aplikace jakožto Youtube a Facebook oficiálně Flash ze svých portálů odstraňují. Též Adobe otevřeně přiznává, že Flash je postupně vytlačován HTML5 – nicméně součástí webů bude do doby, než dojde k doladění všech nastupujících technologií.

Pro moderní web, který používá HTML5 jako náhradu Flashe, je přínosem například to, že video na Youtube (kde je plně integrováno zobrazení přehrávače videa pomocí HTML5) je uživatel schopný na mobilním zařízení spustit přímo ve stránce mobilního webového prohlížeče a nepotřebuje tak žádnou externí aplikaci.

1.4.3 Různé platformy

Moderní web dnes musí počítat s tím, že na něj uživatelé budou přistupovat z různých platform. Dříve zpravidla stačilo web otestovat na třech hlavních webových prohlížečích a případně doladit některé chyby vznikající z různé úrovně podpory webových standardů. (Zde byl problém hlavně s Internet Explorerem integrovaným ve Windows, kde každá verze interpretovala standardy odlišně – tento fenomén je od verze 10 již našťěstí minulostí).

Dnes je však situace jiná. Moderní webové stránky musí počítat s tím, že budou zobrazovány na:

- Klasických notebooků a desktopech
- Mobilních zařízeních
- Chytrých televizích a herních konzolích
- Zařízeních pro Internet věcí

Každá platforma používá odlišné webové prohlížeče založené na různých vykreslovacích jádrech. Vzhledem k velikosti zobrazovacích zařízení je nutné počítat s širokou škálou podporovaných rozlišení. Zatímco klasické notebooky a mobilní zařízení používají standardní operační systémy, u chytrých televizí každý výrobce používá svůj vlastní

system se svým webovým prohlížečem. Proto může být v tomto případě optimalizace poněkud náročnější. U zařízení pro Internet věcí je třeba počítat s tím, že tato zařízení budou vyžadovat specifické komunikační rozhraní pro předávání obsahu na web. V současné době však nejsou pevně definovány upřesňující standardy pro tato zařízení a bude tedy nutné sledovat, jakým směrem se jejich vývoj vydá.

Ačkoliv dnes vývojáři mají práci ulehčenou širokou škálou vývojářských nástrojů pro vykreslení webů pomocí různých rozlišení a vykreslovacích jader, je při vývoji moderního webu důležité pečlivě zvážit návrh celého řešení tak, aby bylo možné jej zobrazit a provozovat na různých platformách

1.4.4 Javascript, JQuery

Součástí moderního webového portálu by samozřejmě mělo být použití JavaScriptu. Od představení HTML5 se na JavaScript začali vývojáři koukat jako na jazyk vhodný pro tvorbu komplexních aplikací a proto nad ním začali vznikat rozšiřující API. Právě vznikající API se zasloužila o rozšíření JavaScriptu v tak velkém měřítku. Cílem těchto API je zjednodušit práci s JavaScriptem a umožnit používání již připravených funkcí. Dle statistiky W3Techs (2016) využívá alespoň jednu JavaScriptovou knihovnu 72.9 % webových stránek, přičemž největší podíl má knihovna JQuery (70 %). [15], [16]

Knihovna jQuery se dnes stala nepsaným standardem při tvorbě moderních webových stránek. Její vývoj byl zahájen v roce 2006 Johnem Resigem a dodnes je stále rozšiřována a aktualizována (aktuálně je k dispozici verze 2.2.3). V rámci jediného souboru vloženého do stránky je možné používat široké množství funkcí, které jsou díky snadné syntaxi a široké kompatibilitě napříč systémy snadno použitelné. O popularitě jQuery svědčí to, že nad ní vzniklo široké množství rozšiřujících modulů. [17]

Alternativou k jQuery jsou například AngularJS, vyvíjený firmou Google pro zjednodušení tvorby JavaScriptových aplikací, či ReactJS, vyvíjený Facebookem pro tvorbu náročných aplikací využívajících velký počet elementů stránky.

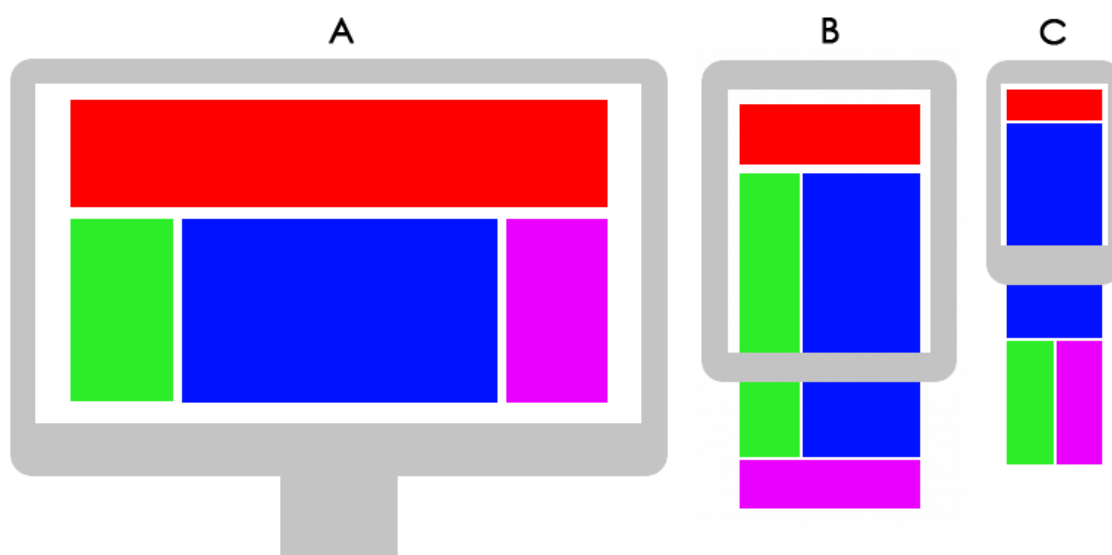
Nad JavaScriptem a jeho moderními API je dnes postavena široká škála knihoven, v následujícím výčtu autor zmíní některé vybrané knihovny, se kterými se seznámil hlouběji a které též v rámci různých projektů využívá:

- **jQuery UI (User interface)** – nadstavba jQuery pro přidání prvků uživatelského rozhraní, efektů, nástrojů a vizuálních vzhledů. Výhodou této knihovny jsou již připravené nástroje pro: výběr dat, našeptávání při vyhledávání, dialogová okna či přepínací panely.
- **Datatables** – nástroj pro zpracování tabulek v prohlížeči. Výhodou je zpracování tabulky tak, že je možné v prohlížeči v reálném čase tabulku třídít, filtrovat či v ní stránkovat. Dalším nesporným kladem je existence rozsáhlého API, pomocí kterého lze práce s tabulkou zpracovávat na serveru a prohlížeči pouze odesílat vyfiltrovaná data – tento přístup se používá zejména u tabulek s velkým počtem řádků, či dotazů náročnějších na zpracování.
- **CKEditor** – textový WYSIWYG (What You See Is What You Get) editor umožňující přidání funkcionalit pro formátování textů známých například z nástrojů sady Microsoft Office do webového prohlížeče.
- **Git.js** – JavaScriptová knihovna postavená nad oblíbeným nástrojem pro správu verzí kódu Git.

1.4.5 Responsivní weby

Řešením problému různých rozlišení na různých platformách je způsob responzivního stylování. Ten je založený na principu, že každá stránka má vytvořenou sadu pravidel, podle kterých je schopna v reálném čase přeskupovat svůj obsah tak, aby umožnila co nejpohodlnější prohlížení. Toho lze dosáhnout například změnou velikostí či přeskupováním některých prvků. [18]

Návrh responzivního webu lze vidět na obrázku 1. Displej A představuje zobrazení na klasickém PC, na němž jsou všechny prvky zobrazeny standardně. Při zobrazení na tabletu B jsou již prvky zmenšené a bloky přeskupené. Na mobilním zařízení je obsah nejvíce zmenšen a prvky jsou přeskupeny úplně.



Obrázek 1: Stejný obsah na různých zařízeních

Zdroj: <https://commons.wikimedia.org/wiki/File:Complete.png>

To je možné právě díky vlastnostem CSS3, konkrétně modulu media-queries popsaného v předchozích kapitolách.

Na mobilních zařízeních je nutné počítat s odlišnou logikou ovládání oproti klasickému PC, kde má uživatel možnost trefit se pomocí myši i do malých ovládacích prvků. Na mobilních zařízeních (a dnes také na stále se rozšiřujících dotykových monitorech) je nutné myslet na to, že do ovládacích prvků se uživatel musí trefit pomocí prstů, což může být zejména na malých displejích problém. Proto by mělo být součástí návrhu základní šablony i definování multiplatformního ovládání.

S podporou mobilních zařízení nepřímo došlo také k vyřešení problematiky zobrazení na chytrých televizích, které jsou limitovány ovládáním pomocí kurzorových šipek na ovladači (popř. pomocí malého touchpadu u dražších zařízení). Větší ovládací prvky jsou přes kostrbatější ovládání pomocí ovladače snadněji použitelné.

Základní přístupy k tvorbě responsivních webů jsou dva: adaptivní a responsivní. Adaptivní přístup definuje několik konkrétních šablon pro zvolená rozlišení, přičemž server posílá do zařízení jen konkrétní verzi dle jeho možností. Naproti tomu responsivní přístup pracuje s jedinou šablonou, která dokáže svůj obsah dynamicky přeskupit dle

aktuálního rozlišení. Výhodou responsivního přístupu je rychlé přizpůsobení obsahu stránky, které je však na úkor datové zátěže – ta je oproti adaptivnímu přístupu vyšší. [1]

Responsivní weby jsou kritizovány, zejména proto, že optimalizovaný web může při nižším rozlišení přicházet o některé funkce. Tato kritika je zcela oprávněná, neboť je možné, že při odbyté optimalizaci může k omezení funkčnosti dojít. Proto by při tvorbě responsivního webu měl být kladem důraz na důkladné testování a tvorbu analýz. Tyto analýzy by měly obsahovat kritéria určující, v jakých případech budou konkrétní prvky stránky skryty, zároveň je v nich třeba pracovat s otázkou, zda nejsou zvýhodňování uživatelé velkých displejů na úkor uživatelů displejů malých či naopak. [18]

Pravidly pro tvorbu responsivních webů by se měly řídit nejen tvůrci webů, ale i tvůrci mobilních aplikací, což bude zmíněno v kapitole 4.

O významnosti optimalizace svědčí i fakt, že vyhledávač Google při hledání z mobilního zařízení zobrazuje informaci o tom, že web je optimalizovaný („*Mobile-friendly*“, v české lokalizaci „*Optimalizováno pro mobily*“).

1.4.6 Očekávání uživatelů

V návaznosti na předchozí kapitoly je vhodné zaměřit se na to, co vlastně uživatelé od webových portálů očekávají. Každý uživatel má své individuální preference, z čehož logicky plyne, že to, co je důležité pro jednoho, může být naprosto zbytečným prvkem pro druhého. Cílem tvůrce moderního webu by tak mělo být pokud možno vyhovět všem, aniž by omezoval funkcionalitu pro jednu skupinu na úkor skupin dalších.

Některá očekávání jsou stále stejná (uživatelé od zařízení očekávají stále stejné funkce bez ohledu na jejich neustálý vývoj), objevují se však též očekávání zcela nová a pro moderní web důležitá. Při tvorbě moderních webových portálů je tedy nejen nutné vycházet ze základních očekávání, ale zároveň vyhledávat očekávání nová.

Uživatelská očekávání lze rozdělit do několika následujících kategorií:

- a) **Rychlost načítání portálu.** Uživatelé neradi čekají – pokud se stránka nenačte ihned, jejich spokojenost klesá. Tvůrce by se tak měl zaměřit na objem přenášených dat a provést kroky k tomu, aby načítání urychlil. Autor této práce má zkušenost s portály, které jsou optimalizovány buď špatně, nebo vůbec: tyto portály sice

nabízejí spoustu vynikajících funkcí a kvalitních obsahů, ale svým pomalým načítáním uživatele od používání spíše odrazují.

- b) **Přehlednost a vhodný design.** Když uživatel prvně vstoupí na webovou stránku, vytváří si první dojem, který rozhoduje o tom, zda na stránce setrvá a zaměří se na obsah nebo jestli stránku opustí. Proto by webové portály měly svou grafikou cílit na požadované zaměření a podle toho i pracovat s přehledností zobrazených dat. Není větší chyby, než když obsah začne fungovat na úkor grafiky – uživatel pak sleduje vizuálně povedený design, ale nedokáže najít potřebné informace.
- c) **Uživatelský komfort.** Při práci s portálem by měl mít uživatel příjemný pocit. Ideálním cílem je vytvořit všechny prvky pro interakci tak, aby byly pro uživatele intuitivní. Na stránce by nemělo být nic pro uživatele obtěžujícího či matoucího. Obtěžující mohou pro uživatele být například následující jevy: nadměra různých dialogových oken, chyby vytvářející nutnost některé postupy opakovat „dokud se to nepovede“, zmatené ovládání způsobené různě umístěnými ovládacími prvky. Matoucí může být například chybné zobrazení navigace nebo špatně strukturovaný web, v jehož podsekcích se uživatelé snadno ztratí.
- d) **Nadstandardní funkce.** Od konkurence je možné odlišit web poutavou grafikou, propracovaným marketingem nebo právě nadstandardními funkcemi. Webový portál by měl oproti konkurenci nabízet „něco navíc“. Nadstandardními funkcemi rozumíme například propojení mezi různými datovými zdroji tak, aby uživatel nemusel stále dokola vyplňovat stejné informace. Dalším příkladem může být propracovaný notifikační systém či integrovaný chat s podporou.
- e) **Podpora mobilních zařízení.** Uživatelé dnes přestávají rozlišovat, z jakého zařízení na webové portály přistupují. Obsah by měl být přístupný napříč všemi platformami bez zásadních omezení. Jak již bylo zmíněno v kapitole Responsivní weby, uživatel očekává též jistý uživatelský komfort, což je potřeba zohlednit při návrhu.
- f) **Propojení se sociálními sítěmi.** Sociální sítě se dnes stávají takřka „internetem v internetu“. Pokud chce nový portál prorazit mezi mladšími generacemi, měl by rozhodně uvažovat o propojení se sociálními sítěmi. Toto propojení může spočívat například v následujících možnostech: přihlášení se do portálu pomocí aplikačního

rozhraní sociálních sítí, přidávání komentářů ke článkům či samostatná tvorba navzájem prolinkovaných textů.

- g) **Povědomí o novinkách.** V dnešní době je pro uživatele již přežitkem pravidelně navštěvovat stránky za účelem zjištění, zda se něco změnilo – dnešní uživatel chce být automaticky o změnách informován. Tato funkce byla dříve doménou čteček RSS (Rich Site Summary – formátu XML pro odběr novinek z webových stránek), dnes se tato funkcionality přesouvá na sociální sítě.

1.5 Zabezpečení webového portálu

V raných dobách internetu, kdy jej využívala jen malá hrstka uživatelů, bylo jeho používání naprosto bezpečné. V dnešní době jsou však útoky na webové stránky (a také na jejich uživatele) na denním pořádku. Pravidelně se objevují útoky, které se snaží získat různá uživatelská data (přístupové údaje, adresy apod.). Tyto útoky probíhají jak pomocí různých praktik na prozrazení přímo od uživatelů (phishingové útoky), tak pomocí specializovaných útoků cílených na velké databáze dat (hacking). Získaná data jsou pro útočníky velmi cenná - například u špatně navržených databází mohou útočníci získat uživatelské jméno a heslo, což je problém zejména z toho důvodu, že uživatelé často používají jeden email a jedno heslo pro více služeb. Webový vývojáři tak mají před sebou nesnadný úkol, protože musí zajistit dostatečnou míru zabezpečení webových stránek. V následujících kapitolách bude specifikováno, jak dosáhnout požadované úrovně zabezpečení v rámci moderního webového portálu. [20]

1.5.1 Práce s hesly

Problém týkající se získání přístupových údajů třetí stranou zmíněný v předchozí kapitole může webový vývojář řešit tak, že zajistí, aby nejslabším článkem při práci s heslem v rámci webového portálu byl právě uživatel. Vývojář bohužel nemá šanci ovlivnit fakt, že uživatelé zpravidla používají stále tatáž hesla, či že si různě přeposílají přístupové údaje pomocí nechráněných komunikačních kanálů. Dalším problémem jsou slabá hesla, kdy při pohledu na žebříček nejčastějších hesel, zpracovaný společností SplashData a vydaný Jamiem Condliffem, je možné zjistit, že nejpopulárnějšími hesly jsou „123456“ a „password“. Nebezpečnost těchto hesel spočívá v tom, že pokud útočník zná přihlašovací jméno, může se do uživatelského účtu velmi snadno dostat pomocí několika málo pokusů. [21]

I přesto, že uživatel používá silné bezpečné heslo (delší heslo složené z různých velkých a malých písmem a číslic), není zaručeno, že jeho interakce s webovými portály bude bezpečná – stále hrozí, že se autoři webových portálů dopustí chyb. Mezi časté chyby patří například uchovávání dat v nezabezpečených databázích v prostém textu, či nedostatečné zabezpečení v případech, kdy autoři portálů hesla hashují (tento problém se týká využití dávno prolomených šifer - MD5, SHA apod.).

Útočníkovi poté stačí využít bruteforce útok (útok hrubou silou), který spočívá v tom, že se snaží ve velkém množství dosazovat do formuláře různé kombinace znaků. Pokud se útočník dostane k databázi a vidí zahashovaná hesla, má možnost je prolomit pomocí takzvaných duhových tabulek (rainbow tables – tabulky předpočítaných hashů hesel), ve kterých pouze vyhledává zvolený hash. V dnešní době nemají vyhráno ani vývojáři, kteří používají takzvané „saltování“ (doplňování hesla o sérii unikátních znaků tak, aby byl každý hash jedinečný). V této fázi může útočník využít velkého výpočetního výkonu (zejména grafických karet) pro dopočítání duhových tabulek dle použitého saltování.

Další chybou vývojářů může být také to, že hesla zasílají společně s registračními údaji v rámci potvrzujících emailů v prostém textu – pokud se útočník dostane do emailové schránky uživatele, dostane se i k heslu. Problematická bývá i obnova hesla, která často umožňuje útočníkovi změnit heslo ve službě tak, že si toho uživatel ani nevšimne.

Co mohou weboví vývojáři dělat, aby se dokázali vyvarovat chyb, případně jak mohou pomoci ke zvýšení bezpečnosti i ze strany uživatelů?

- **Vynutit zadávání silného hesla**, tedy donutit uživatele, aby jako heslo používal kombinaci znaků: malých i velkých písmen, specifických znaků a též čísel. Pomocníkem pro uživatele by mohlo být případné upozornění, že by měl mít pro každou službu jiné heslo.
- **Používat dostatečné zabezpečení při správě hesel**. V současné době je za nejlepší metodu považován Bcrypt – šifrovací funkce založená na šifře Blowfish. Hlavními dvěma výhodami této funkce je, že každý hash je jedinečný (dvě stejná hesla nebudou uložena stejným hashem), a je zde možnost nastavení délky generování hashe, kdy výchozí hodnota má nastaveno generování 80 ms na jedno heslo –

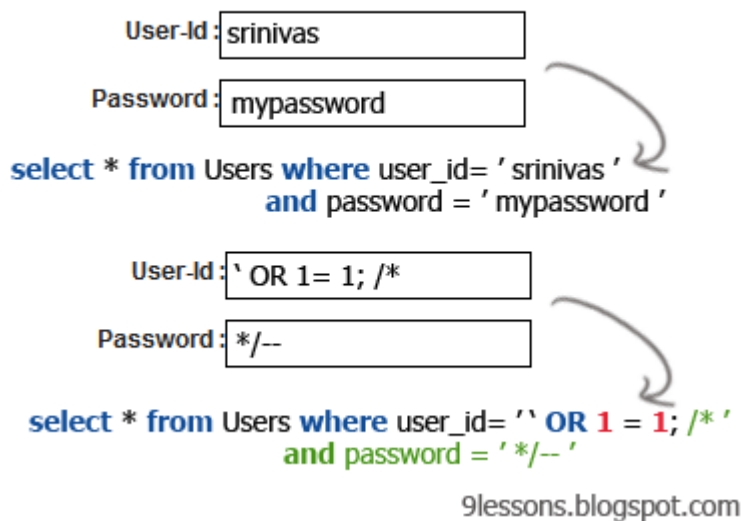
uživatel tuto prodlevu ani nepostřehne, ale případného útočnicka (útočícího zpravidla bruteforce útokem) to výrazně zpomalí, případně od útoku odradí úplně.

- **Obnova hesla bezpečně** – při žádosti o změnu hesla by měl systém vygenerovat uživateli odkaz obsahující hash s limitovanou platností, který již nebude v budoucnosti znovu použit. Při každé žádosti o obnovu hesla by měl také systém automaticky uložit adresu IP a časovou značku. V případě prolomení uživatelského účtu by pomocí těchto údajů mělo být uživateli možné poskytnout alespoň vodítka, kdy a jak k odcizení jeho účtu došlo.

1.5.2 Komunikace s databází

Pro zajištění bezpečnosti při tvorbě moderních webových portálů by neměla být vůbec opomenuta otázka zabezpečení komunikace s databází. Pokud nedojde k její bezpečné realizaci, může potenciální útočník využít jednoduché zranitelnosti - Denny Cherry uvádí, že jednou z nejjednodušších, ale zároveň nejméně ošetřovaných, forem útoků na webové stránky jsou útoky typu SQL Injection. Princip tohoto útoku spočívá v tom, že útočník využije formulářový prvek na stránce či adresu URL k tomu, aby podsunul skriptu zpracovávajícímu vykreslení webové stránky své vlastní instrukce. Rozsah těchto instrukcí může být od jednoduchých výpisů (kde je cílem zobrazení skrytých sloupců v databázi), přes vkládání vlastních údajů do tabulek až k možnosti kompletního smazání databáze. Pokud se útočnickovi povede najít takto zranitelné místo, může také například postupným zkoušením odhalit kompletní strukturu a vnitřní logiku databáze. [22]

SQL Injection.



Obrázek 2: Příklad SQL Injection

Zdroj: <http://cracksoftwaretestinginterviews.blogspot.cz/2014/02/35-sql-injection.html>

Podsunutí instrukce může proběhnout například dle obrázku 2, kde je zobrazen přihlašovací formulář. Uživatel zadává do formuláře své uživatelské jméno („User-Id“) a heslo („Password“). Následný dotaz pak ověří shodu zadaného uživatelského jména a hesla se záznamy v databázi. Pokud dotaz nenalezne shodu, je uživateli zobrazeno hlášení o chybě. V případě útoku útočník do pole uživatelského jména dopíše vlastní instrukci „\ OR 1 = 1; /*“ a do pole heslo instrukci „*/--“, čímž při zpracování proběhne změna dotazu tak, že podmínka zní „nalezni vše z tabulky uživatelé, kde je uživatelské jméno prázdné, nebo kde 1=1“. Případné zbylé podmínky jsou tímto zápisem zrušeny a díky doplněné podmínce „kde 1=1“, která bude vždy splněna, se útočník může snadno přihlásit do administrace bez znalosti přístupových údajů.

Řešení tohoto problému je dvojí, zaprvé pořádné ošetření vstupů, zadruhé využití předpřipravených instrukcí. Zabezpečení na bázi ošetření vstupů se snaží u každého možného uživatelského vstupu zajistit jeho nezneužitelnost. Číselné proměnné jsou před zpracováním převáděny na číselné datové typy a ověřované. U textových vstupů je situace složitější v tom, že je zapotřebí hlídat, zdali se ve vstupu nevyskytují zakázané znaky (uvozovky, hvězdičky, apod.). Zabezpečení pomocí předpřipravených instrukcí vypadá tak, že v dotazu do databáze se namísto dosazování proměnných dosadí zástupné parametry. Těmto parametrům jsou následně pomocí dalších funkcí přiřazeny hodnoty.

Výhodou tohoto řešení je, že je vstup ošetřen automaticky pomocí rozhraní pro komunikaci s databází. Není proto možné spustit žádný odlišný kód. V jazyce PHP jsou tato rozhraní realizována například pomocí doplňujících knihoven PDO a MySQLi, kde PDO lze použít nejen pro MySQL databáze. Prostředí ASP.NET má již rozhraní pro předpřipravené instrukce integrované.

1.5.3 HTTPS

Komunikace mezi webovými stránkami probíhá pomocí HTTP (Hypertext Transfer Protocol). Tento protokol však není vůbec zabezpečený - potenciální útočník může sledovat komunikaci mezi webovým prohlížečem a cílovým serverem. Při sledování komunikace může útočník číst veškerou nezašifrovanou komunikaci a získávat z ní citlivé uživatelské informace, či podstrčit uživateli vlastní stránku. Tato stránka například může vyžadovat fiktivní „kontrolní“ ověření údajů s tím, že po zpracování údajů přesměruje uživatele na cílovou stránku a ten tak ani nepozná, že se stal obětí. [4]

Proto by měl každý webový portál, který od uživatele vyžaduje jakékoliv citlivé informace, používat HTTPS (kde S znamená Secured - zabezpečený). HTTPS přenáší informace mezi prohlížečem a serverem v šifrované formě, takže není možné jej odposlechnout. Současně také zajišťuje důvěryhodnost komunikace tím, že prohlížeč stále ví o probíhající komunikaci s ověřeným serverem a není mu tak podstrčena žádná falešná stránka.

Pro zavedení a provozování tohoto zabezpečení je pro každého vlastníka webového portálu nutné vlastnictví důvěryhodného SSL certifikátu, který musí být podepsán důvěryhodnou certifikační autoritou.

1.5.4 Uživatelská práva u databází

Pro komunikaci s databází by měly existovat minimálně dva účty – jeden pro administrátora s právy na veškerou administrativní činnost spojenou s databází (což může být limitováno webhostingem, kde tyto účty mají například zakázáno vytváření vlastních databází) a druhý pro uživatele, který nemá možnost provádět zásadní operace nad tabulkami (vytváření či nastavování klíčových parametrů, odstraňování tabulek apod.). Administrátorský účet by měl být využíván pouze vytváření databáze a její správu. Nikdy by se neměl objevit ve zdrojovém kódu aplikace. Pro aplikaci je vhodné využít právě uživatelský účet. Zajistí se tím tak, že při případném útoku SQL injection, či zjištění

přístupových údajů k tomuto účtu, nebude moci útočník provést zásadní škody v databázi - pokud tedy jako zásadní škodu nepočítáme ztrátu citlivých osobních informací uložených v databázi

Další výhoda nastavení uživatelských práv vynikne při používání procedur. V tomto případě má aplikační účet práva pouze na čtení z tabulek (ne na zápis, modifikace či mazání) a na spouštění procedur (v rámci kterých je zápis, modifikace, či mazání řešeno). V tomto případě je potenciální útočník limitován při provádění změn v databázi pouze na použití procedur, jejichž strukturu a funkčnost nezná. To jej zpravidla může odradit a zajistit tak vyšší bezpečnost webového portálu.

1.5.5 Cíle pro moderní web

Vzhledem k popsaným typům útoků je bezpečnost v rámci webových portálů velice důležité téma. Vývojové týmy by měli klást důraz na to, aby každý vyvíjený portál počítal se všemi (i méně pravděpodobnými) alternativami jeho používání a potenciálně zneužitelnými slabými místy. Součástí vývoje by mělo být i důkladné ošetření všech vstupů. Během vývoje a následného testování by v ideálním případě nemělo dojít ke vzniku žádných bezpečnostních chyb. Pokud v kódu portálu chyby vzniknou, měly by být urychleně opraveny a celý proces vývoje zdokonalen tak, aby k podobným chybám v budoucnosti dojít nemohlo. Současně by měl vývojový tým sledovat aktuální vývoj nejen technologií pro zabezpečení dat a webových portálů, ale také i technik použitelných pro potenciální útočníky. Pokud jsou součástí portálů technologie třetích stran, měly by být pravidelně (po náležitém otestování) aktualizovány. Denny Cherry doporučuje zvolit při zabezpečování webových portálů takový přístup, který raději počítá s útokem, jenž nemusí reálně nastat, nežli řešit složité nápravy škod po zrealizovaném útoku. [23]

1.6 Databáze

Jak již bylo v předchozích kapitolách zmíněno, databáze jsou nástroje pro správu dat. V rámci jednoduchých webových prezentací zpravidla není důvod k jejich využití. Avšak u webových projektů, které počítají se vznikem většího množství dynamických dat je téměř nezbytné jejich použití. Chris Auld a kolektiv uvádějí, že hlavní přínosy vyplývající z použití databází jsou [24]:

- Data jsou setříděná a strukturovaná

- Více uživatelů může pracovat najednou se stejnými daty
- Je snadné mezi daty vyhledávat a zkoumat jejich relace
- Je možné data exportovat a provádět nad nimi statistické úlohy
- Vzniká možnost pevně určit vztahy mezi daty

V rámci tvorby moderního webového portálu budou v kapitole 3 využity vybrané pokročilejší přístupy k tvorbě databází. Těmito přístupy jsou tvorba procedur a funkcí, tvorba databázových pohledů a využití transakcí.

1.6.1 Procedury a vlastní funkce

Procedury existují, aby ve zdrojovém kódu nebylo potřeba používat příkazy SQL pro manipulaci s daty (INSERT, UPDATE či DELETE), a zároveň, aby aplikační uživatel nemusel mít práva na spouštění těchto příkazů (jak již bylo zmíněno v kapitole 1.5.4 Uživatelská práva u databází). Procedury jsou předpřipravené SQL kódy pro práci s databází a záleží jen na vývojáři, jakou funkcionalitu jim naprogramuje. Výhodou procedur je, že nemohou spustit žádný jiný kód, než jaký mají v sobě integrován, tj. není možné jim podsunout žádný škodlivý kód. Do procedury může být také zahrnuto automatické ukládání informací do logovacích tabulek. Pomocí těchto tabulek je možné zpětně zjistit kdo, kde a jaké úpravy v databázi prováděl. Nevýhodou procedur je, že jsou náročnější na tvorbu a že v rámci každého databázového řešení (např. MySQL a MSSQL) je jejich syntaxe odlišná. [22]

Další důležitou funkcionalitou v rámci pokročilejších přístupů ke tvorbě databází je možnost vytváření vlastních databázových funkcí pro práci s daty. Oproti procedurám je funkce možné spouštět v rámci dotazu do databáze (SELECT). Funkce buď může mít specifikované parametry (například funkce pro výpočet ceny bez DPH by mohla mít parametry cena s DPH a daňová sazba), nebo žádné parametry mít nemusí (například funkce pro výpis čísla aktuálního týdne). U MSSQL mohou funkce vracet dokonce i tabulky. Z hlediska oprávnění stačí uživatelskému účtu databáze nastavit pouze práva na výběr (SELECT) z těchto funkcí.

1.6.2 Pohledy

Pokud se v kódu webového portálu na různých místech opakují stejné složitější dotazy do databáze, je vhodné použít virtuální databázové struktury - takzvané pohledy. Tyto

pohledy jsou vhodné pro zjednodušení a i zpřehlednění práce s databází. Jejich použití může být vhodné například v situacích, kdy je potřeba:

- Zpřístupnit uživatelskému účtu databáze jen určitou část některé tabulky
- Použít různé databázové funkce či vlastní výpočty – například vypočítávání součtů a průměrů dat v tabulce
- Sloučit více tabulek do jedné (např. pomocí příkazů JOIN či UNION)

1.6.3 Transakce

Použití transakcí umožňuje vrátit databázi do jejího předchozího stavu při vzniklé chybě během zpracování dat. V rámci zpracování zdrojového kódu může při vykonávání složitější operace¹ skládající se z více dílčích úloh dojít v rámci jedné z těchto úloh k chybě. Pro opravu chyby v klasickém případě (tedy v případě, kdy programátor nevyužívá transakce) je nutné všechny změny v databázi zpětně dohledat a napravit. V případě použití transakcí však stačí pouze zrušit transakci a databáze se vrátí do stavu, v jakém byla před zahájením zpracování celé operace. Pokud vše proběhne v pořádku, stačí transakci pouze potvrdit.

1.7 Plán vývoje

Před zahájením vzniku každé vyvíjené komponenty (webového portálu, aplikace, informačního systému, doplňkového modulu apod.) je důležité definovat její životní cyklus. Tento cyklus začíná posouzením, zda je potřeba komponentu vůbec vytvářet, a končí ukončením provozu; lze jej rozdělit do jednotlivých etap dle použité metodiky.

Dnes existuje velké množství různých metodik, přičemž každý vývojový tým může používat jejich různě přizpůsobené verze (které mohou vycházet z firemní politiky, rozložení týmu či osobních preferencí). Tyto metodiky se zpravidla liší různou interpretací, slučováním a rozdělováním jednotlivých etap životního cyklu. [25]

Pro vývoj moderního webového portálu v praktické části této práce bude použita metodika System Development Life Cycle (SDLC), s níž má autor nejlepší zkušenosti ze svého působení na praxi v podniku i z participace na dalších mimoškolních projektech. SDLC je

¹ Příkladem takové složitější operace může být zpracování objednávky v eshopu, při němž se do databáze nejdříve uloží informace o uživateli, následně se zpracují data v nákupním košíku a poté dojde k založení objednávky.

životní cyklus projektu vytvořený přímo na míru pro tvorbu, úpravy a údržbu komponent. [26]

Alternativou k SDLC může být například Rapidní vývoj aplikací (Rapid Application Development), který spočívá v rychlém vývoji komponenty pomocí paralelního rozdělení projektu do jednotlivých funkčních sekcí. Výhodou tohoto řešení je, že vývojový tým je schopen od zákazníka velice rychle získat zpětnou vazbu z hotového prototypu. Nevýhoda vzniká tím, že je k jejímu nasazení zapotřebí silný a zkušený tým se zkušeným manažerem řídícím celý vývoj.

Další alternativou může být metodika SCRUM, která se na celý vývoj komponenty dívá jako na jednotlivé, na sebe navazující kroky. Základní myšlenkou této metodiky je, že požadavky na vyvíjenou komponentu se budou v průběhu tvorby lišit, a je tedy potřeba vývoj rozdělit na pevně ohraničené části. Proto je tato metodika vhodná při tvorbě projektů, u kterých je sice jasná vize cílové komponenty, ale nejsou finálně vymyšleny cílové požadavky a spíše se předpokládá, že se tyto požadavky budou v průběhu vývoje měnit.

Oproti těmto metodikám přináší SDLC výhody v tom, že v rámci celého procesu je každá fáze pevně zdokumentovaná a ohraničená. V případě potřeby změny v jedné z navazujících fází je celý proces zopakován tak, aby byl výsledný produkt kompletně odladen. Problémem mohou být zvýšené časové nároky na realizaci jednotlivých kroků a také vyšší nároky na manažera řídícího celý proces.

Metodika SDLC je složena z následujících devíti etap:

1.7.1 Plánování cílů

První krok zahajující životní cyklus komponenty. V této fázi je nutné zvážit, zdali je potřeba vznik nové komponenty. Dochází zde k nastínění hrubých odhadů a analýz, zda potřebné vynaložené náklady (peněžní, lidské či časové) zajistí do budoucna realizaci požadovaných cílů. Následně jsou tyto cíle transformovány na seznam požadavků a funkcí. V této fázi by také mělo dojít ke zvážení integrace s dalšími prvky v rámci IS zadavatele (firemní systémy, databáze, aplikace apod.)

Při úvahách o tvorbě nové komponenty, která má nahradit stávající, již používanou komponentu, je důležitou součástí této specifikace i popis současného stavu a definice důvodů pro tvorbu nové komponenty.

Posledním krokem před dokončením specifikace cílů jsou rámcové analýzy možností vývoje, odhady časových náročností a nákladů spojených s pořízením i provozem komponenty (údržba, servis, distribuce apod.).

1.7.2 Plánování požadavků.

V této fázi je již rozhodnuto, že dojde k vývoji nové komponenty. Nyní dochází k přesnému návrhu systému včetně všech jeho podrobností. Návrh ze specifikace cílů je pečlivě analyzován a detailně rozpracován, provádí se tvorba různých diagramů a modelů. [27]

Těmito modely jsou např.:

- **Use Case diagram** (diagramy užití) určující jednotlivé uživatelské role a procesy v rámci komponenty
- **Diagram aktivit** popisující aplikační logiku a jednotlivé procesy v rámci komponenty

Probíhá zde také stanovení požadavků na uživatelské rozhraní, definují se aplikační postupy a určuje se předpokládána práce uživatelů s komponentou.

Uživatelské role, určené pomocí diagramu užití, jsou doplněny o specifikace určující, do kterých sekcí komponenty bude mít uživatel přístup a také jaké povinnosti bude mít v rámci ovládání těchto sekcí.

Na konci této fáze se provádí specifikace hardwaru a softwaru, na kterém bude komponenta nasazena a provozována. Součástí těchto specifikací je i určení, zda bude komponenta podporovat mobilní zařízení (tj. zda bude vyžadováno, aby byla data zobrazitelná na malých displejích) či zda bude v rámci komponenty vznikat i mobilní aplikace. U projektů, které začaly vznikat bez podpory mobilních zařízení, a až v některé pozdější fázi (i ve fázi rutinního provozu) bylo o jejich podpoře rozhodnuto, hrozí, že bude zapotřebí provést zásadní změny v koncepci hlavní komponenty. To znamená, že bude zapotřebí absolvovat znovu všechny kroky vývoje (což s sebou nese více nákladů).

Při specifikaci SW dochází i k volbě vhodného vývojového a databázového prostředí, programovacího jazyku a případných dalších rozšiřujících komponent.

Společně se specifikací hardwaru se plánují i potřebné přenosové kapacity (objem odesílaných a stahovaných dat, množství vznikajících požadavků na data apod.). Také probíhá definování případného API a specifikování komunikačních protokolů či automatizačních úkonů (např. periodické spouštění skriptů či mechanismů pro očištění a zálohování dat.).

1.7.3 Návrh zvoleného řešení

Jakmile jsou určeny cíle a požadavky, je možné zahájit reálnou tvorbu komponenty. Zahájením tvorby by měla být porada, během níž se sejde celý vývojový tým (programátoři, grafici a hlavní testeři) s vedoucími projektu. Během této porady jsou prodiskutovány způsoby realizace určených cílů a požadavků a rozděleny úkoly mezi jednotlivé členy dle konkrétně zvolené metodiky a pracovní struktury.

Jakmile jsou vývojáři seznámeni s projektem, dochází k tvorbě databázových struktur, určování relací mezi jednotlivými tabulkami a k tvorbě procedur a pohledů. Nad vznikající databází je následně možné zahájit psaní zdrojových kódů komponenty s kladením důrazu na správnou funkčnost a uživatelskou přívětivost.

Grafici v této fázi vytvářejí vzhled celé komponenty (loga, menu, jednotlivé stránky apod.) a společně s vývojáři ladí umístění ovládacích prvků. V rámci komponenty pak vznikají jednotlivé formuláře a datové výpisy. UX (User Experience) designéři začínají vytvářet a ladit firemní identitu. Příprava zobrazovaných textů v této fázi je úkolem copywriterů.

Pokud má být součástí komponenty API, jsou zdrojové kódy vytvářeny tak, aby bylo možné toto rozhraní připravit na budoucí využití.

Zároveň s komponentou vzniká administrační nástroj, pomocí kterého bude v budoucnu možné komponentu konfigurovat a také spravovat.

V celém průběhu této fáze je vytvářena podrobná kompletní dokumentace, která bude v budoucnu sloužit nejen uživatelům (jako návod pro používání aplikace a popis jednotlivých funkcí komponenty), ale také i administrátorům (je nutné, aby věděli co a kde spravovat v rámci svých rolí) a vývojářům (pro znalost fungování celé komponenty).

Pokud se v budoucnu vývojový tým rozhodne komponentu rozšiřovat či modifikovat, nebudou vývojáři muset složitě analyzovat zdrojové kódy, ale pouze nahlédnou do dokumentace. Na základech této vznikající dokumentace je postupně rozplánováno školení testerů, administrátorů a i uživatelů.

Před tím, než je možné tuto fázi ukončit, musí celý vývojový tým provést podrobné testy funkčnosti (vývojáři nesmí předávat nefunkční kód). Každá chyba nalezená v této fázi musí být pečlivě zanalyzována a opravena tak, aby již znovu nevznikla. Jakmile jsou testy úspěšně splněny, je možné přejít k fázi implementace.

1.7.4 Implementace do testovacího prostředí

Po ukončení vývoje je čas pro nasazení komponenty do testovacího prostředí (ať už se jedná o prostředí v rámci zadávající firmy či o veřejně nepřístupnou část webového hostingu). Cílem této fáze je ověření, jak bude komponenta fungovat v prostředí velice podobném tomu, ve kterém bude provozována. Tato fáze se provádí zejména proto, že vývojové prostředí (např. server vývojového týmu) může mít odlišnou konfiguraci (popř. jiná specifika) oproti produkčnímu prostředí.

Souběžně s tím je zpřístupněno API pro další systémy a jeho funkčnost je pečlivě monitorována. Komponenta je naplněna testovacími daty (to mohou být data reálná či vygenerovaná) a texty od copywriterů. Nad těmito daty je možné provést přesné doladění na míru testovacího prostředí. Poté je zapotřebí pomocí vzniklé dokumentace provést školení testovacích uživatelů a i případných klíčových uživatelů (hlavních uživatelů, kteří budou aplikaci používat a budou zodpovědní za její chod).

1.7.5 Celkové testování

V této fázi je komponenta předána testujícím a klíčovým uživatelům s cílem otestování veškeré funkčnosti předtím, než bude možné komponentu nasadit do produkčního prostředí. Cílem „testerů“ je odhalit, zda v rámci komponenty neexistují zatím neodhalené chyby a zda komponenta přesně odpovídá zadání a dodané dokumentaci. Během testování vznikají zprávy dokumentující průběh a výsledky testování.

Z hlediska testování jsou prováděny dva základní typy testů [28]:

- **Zátěžové a výkonnostní testy.** Při nich je celá komponenta podrobena testům, které se snaží o její plné vytížení. Testuje se například vysoká zátěž databáze, množství paralelně prováděných operací či vytížení API a komunikačních kanálů. Souběžně se testuje také výkonnost komponenty (rychlost načítání dat, přenosové kapacity, analýzy přenášených dat).
- **Testy uživatelské přívětivosti.** V rámci těchto testů je pečlivě zkoumána ergonomie a intuitivnost při používání komponenty uživateli. Výstupem těchto testů by mělo být hodnocení, jak pohodlně se s komponentou pracuje a jak je její ovládání náročné.

V průběhu testování jsou pravidelně realizovány schůzky celého vývojového týmu s testery a se zadavatelem komponenty, během kterých jsou plánovány testovací scénáře a diskutovány výsledky již proběhlých hodnocení. Pokud jsou v této fázi zjištěny zásadní nedostatky, je komponenta vrácena zpět do fáze návrhu s cílem opravení chyb a odladění opravného balíčku. Pokud jsou provedeny a akceptovány všechny testovací scénáře, je možné tuto fázi ukončit.

1.7.6 Zavádění systému

Komponenta, která se dostane do této fáze, je již připravena na přechod do produkčního prostředí. Testovací data jsou vymazána a nahrazena daty produkčními. Veškerá napojení na testovací prostředí jsou zrušena a komponenta je přepnuta do režimu komunikace s produkčními systémy. Pokud byla v rámci testování zapnuta ladící rozhraní (výpis dotazů dat, podrobné chybové hlášky apod.), v této fázi jsou vypnuta.

Zároveň se zaváděním probíhá kompletní školení všech uživatelů a administrátorů.

1.7.7 Zkušební období

V této fázi je zahájen provoz komponenty v produkčním prostředí. Pokud není určeno jinak, je tato fáze pevně ohraničena časovým úsekem (v rádech dnů až týdnů). V rámci této fáze je provoz komponenty pečlivě monitorován tak, aby byly odhaleny potenciální chyby. Součástí monitoringu může být sledování reálného vytížení komponenty nejen uživateli, ale i ostatními systémy. Pro vývojové týmy je nutné počítat s tím, že musí být připraveny v co nejkratší možné době poskytnout uživatelům servis a případně i vysvětlit principy fungování (zde se naplno využívá vzniklé dokumentace).

Jakmile je zkušební období ukončeno, je na čase provést finální zhodnocení výsledné komponenty. Pokud jsou klíčoví uživatelé a zadavatel se systémem spokojeni, je možné systém předat do fáze rutinního provozu. Jsou-li nalezeny závažné nedostatky bránící v akceptaci, je nutné vrátit systém zpět k návrhu a celý proces vývoje zopakovat.

Pokud cílem nově vyvíjené komponenty mělo být nahrazení stávající komponenty, je možné provoz původní komponenty ukončit.

1.7.8 Rutinní provoz a údržba

V rámci rutinního provozu je komponenta běžně používána a probíhají nad ní pouze běžné úkony údržby. Zaměstnanci využívají její funkce. Administrátoři se starají o ovládání komponenty a správu jejích dat. Stávající uživatelé a administrátoři jsou také schopni proškolit nové uživatele.

Pokud je potřeba provést zásadnější změnu v rámci komponenty nebo je nalezena chyba, je zapotřebí zahájit znovu celý cyklus SDLC, kdy se buď upraví stávající komponenta, nebo je vyvinut rozšiřující modul nebo servisní balíček.

1.7.9 Zásadní přehodnocení funkčnosti

Jakmile komponenta neodpovídá požadavkům zadavatele (je potřeba provést její zásadní úpravu, dojde k ukončení podpory výrobcem či se zásadně změní legislativa nebo vnitřní normy), je nutné zvážit vývoj jejího nástupce. V této fázi jsou provedeny kroky, které mají za cíl kompletně analyzovat a určit, jaká byla současná komponenta a jak by měl vypadat její následovník. Vývoj nové komponenty je poté zahájen. Jakmile dojde její životní cyklus do fáze akceptace testovacího provozu, je možné současnou komponentu vypnout a tím i ukončit její provoz.

2 Analýza vybraného projektu

V této části bude analyzován vývoj reálně uskutečňovaného start-up projektu Reklamář.cz. Projekt je realizován autorem této práce a jeho kolegou z Technické univerzity v Liberci. Autor v rámci projektu zastává pozici hlavního vývojáře a dohlíží na jeho realizaci tak, aby byly splněny podmínky tvorby moderního webového portálu. S realizací vypomáhá tým složený z externího poradce, konzultanta grafiky a UX designéra a několika nezávislých testerů. Tento tým byl sestaven na základě dobrých předchozích zkušeností z realizace několika dalších webových projektů, kdy jeden se například zabýval vývojem a provozováním slevového portálu BankaSlev.cz.



Obrázek 3: Logo portálu Reklamář.cz

Zdroj: vlastní

2.1 Základní koncepce (plánování cílů)

Cílem toho projektu je realizace moderního online webového portálu pro evidenci nákupních dokladů (účtenek, faktur apod.) v elektronické formě. Tato evidence by měla umožňovat jednoduchou správu vložených produktů. Zároveň by měla obsahovat automatický nástroj pro hlídání blížícího se termínu expirace doby pro možnou reklamaci produktů. Současně s tím je naplánována funkcionalita s úkolem pomáhat uživatelům s reklamacemi – uživatel bude mít možnost pomocí integrovaného nástroje zahájit reklamační proces, následně monitorovat průběh reklamace (s přihlédnutím k zákonem daným lhůtám). V případě nespokojenosti s průběhem reklamace by měl mít uživatel možnost zjistit, jak postupovat. Pro poskytnutí kompletní podpory návštěvníkům webu by měla být vytvořena znalostní báze, ve které budou zodpovězeny všechny nejčastější otázky týkající se nákupu produktu, reklamací a obecných práv a povinností zákazníka. Jako poslední část portálu je plánován modul sloužící k hodnocení firem dle uskutečněných reklamací. Projekt je zamýšlen jako alternativa k srovnávačům zboží Heureka.cz či Zboží.cz, které se reklamačním procesem nezaobírají.

2.1.1 Důvod vzniku portálu

Tento projekt je v současné době na českém internetu unikátní tím, že řeší problematiku reklamací, kterou se nikdo komplexně nezaobírá. Při pokusu o vyhledání informací ohledně reklamací („*jak reklamovat*“, „*vrácení zboží*“, „*reklamace*“) chybí jednotné místo s ověřenými informacemi. Uživatel mnohdy končí u neověřených příspěvků v prostředí různých diskuzních fór či na zpravodajských portálech typu Idnes.cz a Novinky.cz, kde nemusí být poskytnuté informace vždy aktuální.

Současně na trhu není k dispozici komplexní nástroj pro práci s účtenkami, který by dále řešil termíny expirací doby případné reklamace a následný reklamační proces. Na trhu se sice objevuje několik projektů, které umožňují nahrávání účtenek, ty ale slouží pouze jako úložiště souborů. Některé řetězce prodejen typu Hervis a Okay mají v rámci svých věrnostních programů automaticky propojené doklady s jednotlivými nákupy, nicméně v těchto případech se jedná pouze o propojení účtenek a zboží zakoupeného v jejich sítích obchodů.

V době psaní této práce se na trhu objevil produkt, který by se teoreticky dal označit za potenciální konkurenci – nástroj Šanon, který je součástí elektronického bankovníctví od bankovní společnosti AirBank. Tento nástroj je však přístupný pouze vlastníkům bankovního účtu AirBank a od konce roku 2016 je plánováno jeho zpoplatnění. Z hlediska funkčnosti je pomocí tohoto nástroje možné nahrávat účtenek a hlídat termíny, avšak chybí zde nástroj pro realizaci případných reklamací. [29]

2.1.2 Současný stav

Současný stav je takový, že zakoupí-li běžný zákazník produkt, musí si sám hlídat lhůtu, v jejímž rámci je možné produkt reklamovat. Dle zkušeností z autorova okolí si lidé datum pamatují, případně si jej napíší do kalendáře. Často však spíše na datum zapomenou a v případě problému doufají, že je lhůta pro reklamaci stále aktivní. Současně platí, že zákazník je povinen mít pro možnost případné reklamace účtenku uloženou, ideálně na bezpečném místě.² Při koupi a případné reklamaci běžný zákazník nezná svá práva a povinnosti a může tak docházet ke spoustě omylů. Celý proces reklamací je pro běžného zákazníka velkou neznámou – reklamace produktů není činnost, kterou by dělal denně a

² To může být například libovolný šanon, ale je možné i pokročilejší řešení – naskenovat dokumenty do PC.

měl s ní větší zkušenosti. Reklamář by měl zákazníkům různých obchodů tuto zkušenost poskytnout: poradit jim jak reklamovat a pomoci jim porovnat své zkušenosti se zkušenostmi ostatních zákazníků s reklamacemi u různých prodejců.

2.1.3 Plán realizace

Pro vznik tohoto moderního webového portálu byla zvolena metodika SDLC popsaná v předchozích kapitolách. Před jejím započítím byly zváženy možnosti týmu tvůrců. Ten byl pro dosažení co nejlepšího možného řešení sestaven z následujících členů:

- **Projektový vedoucí a vývojář** – autor této diplomové práce a spoluautor myšlenky vzniku portálu. Cílem vývojáře bude, jak již bylo zmíněno výše, zajistit tvorbu moderního webového portálu s přihlédnutím k mobilním zařízením. Portál bude optimalizován pro mobilní zařízení, jeho součástí bude i mobilní aplikace. V rámci role projektového vedoucího bude autor zodpovědný za kompletní technickou stránku vývoje, nasazení a provoz tohoto projektu.
- **Marketing & expert** – spoluautor myšlenky vzniku portálu, jehož cílem bude vytvoření marketingových kampaní pro připravovanou propagaci na internetu. Zároveň také bude zastávat roli SEO experta – zaměří se na úpravu koncepce portálu tak, aby byl dobře indexovatelný webovými vyhledávači. Další úkoly tohoto člena se týkají optimalizace grafických prvků, analýzy funkčnosti portálu a optimalizace textů.
- **Externí poradce** – poradce, se kterým měli tvůrci dobré zkušenosti z předchozích projektů. S tímto členem budou v průběhu tvorby prodiskutovány aspekty technické stránky vyvíjeného řešení.
- **Grafický konzultant** – člen, se kterým bude projekt konzultován tak, aby bylo možné plně využít všech aspektů grafiky pro mobilní zařízení a splnit nároky na vzhled moderního webu. Tento konzultant má rozsáhlé zkušenosti s realizací velkých webových projektů.
- **Nezávislí testeři** – uživatelé, kteří budou provádět série vybraných testů s cílem maximálního odladění portálů a poskytovat potřebnou zpětnou vazbu pro optimalizaci všech částí portálu.

2.1.4 Odhady časových a finančních náročností

Pro běžného uživatele bude používání všech částí webového portálu zdarma. Aby byl projekt finančně výnosný, bylo rozhodnuto, že příjmy z tohoto projektu budou plynout ze zobrazované webové reklamy (zejména z nástroje Google AdWords) a z možnosti propagace firem (tvorba placených článků, úprava pozic v žebříčcích). V rámci vnitřní projektové politiky nebude firmám umožněno, aby docházelo k vymazávání špatných hodnocení od uživatelů – bylo by to zcela v rozporu se základní myšlenkou tohoto portálu, která je o pomoci uživatelům.

Plánovaná časová náročnost vývoje webového portálu by se měla pohybovat v rámci 100 – 130 hodin. Testování veškerých funkcí by mělo být v minimálním rozsahu 30 hodin.

Z důvodu, že vývoj bude realizován vlastními silami, a nebude využito žádné komerčně dostupné řešení, se výsledná finanční náročnost skládá pouze z nákladů na webhosting (cca 400,- Kč / rok), doménu (151,- Kč / rok) a SSL certifikát (400 – 700 Kč / rok).

2.2 Požadovaná funkčnost (plánování požadavků)

Po zhodnocení současného stavu a též základní koncepce byl projekt označen jako realizovatelný. Bylo tak možné přejít od plánování cílů k plánování požadavků. Cíle byly znovu analyzovány a na jejich základech vznikl plán požadované funkčnosti.

Celý projekt by se měl skládat ze tří hlavních modulů:

- a) **Modul dokladů a reklamací** – modul zabývající se archivací dokladů o koupi, dále správou a zahájením reklamačního procesu
- b) **Modul hodnocení firem** – modul, kde bude možné zobrazit a provést hodnocení firem dle zkušeností návštěvníků webu s reklamacemi
- c) **Modul poradny** – model, který bude sloužit návštěvníkům webu hledajícím rady v problematice nákupu a reklamací

Nad těmito moduly by měla být uživatelská sekce, vyžadující registraci uživatelů pomocí emailů či profilů na sociální síti Facebook.

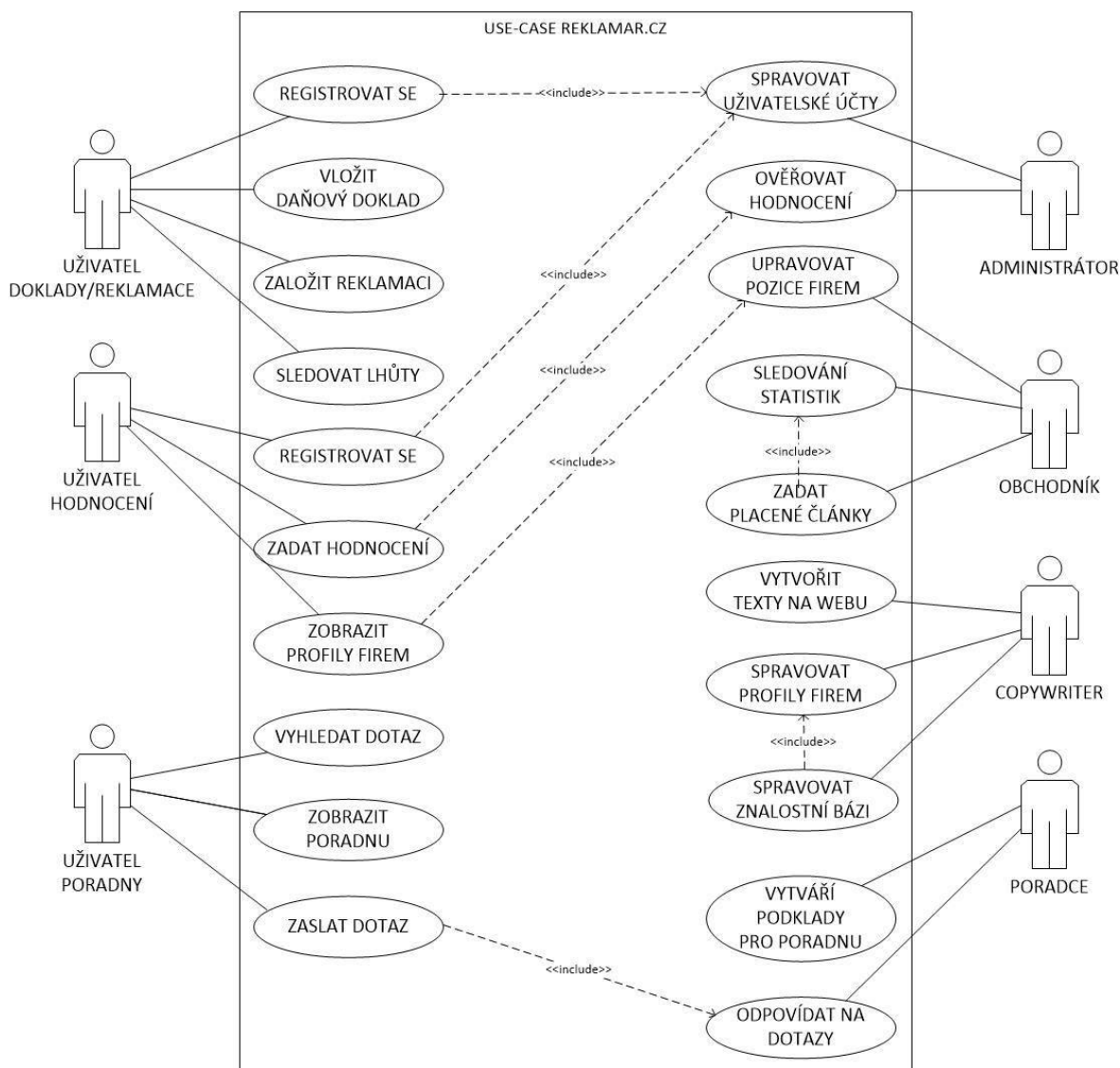
Současně by také měl být k dispozici pokročilý redakční systém pro kompletní správu obsahu portálu v rámci webového prohlížeče. Tento redakční systém by měl také poskytnout nástroje pro optimalizace SEO kampaní.

2.3 Diagramy

Pro zjednodušení vývoje a ujasnění si základních koncepcí byly v této fázi použity vybrané standardizované diagramy.

2.3.1 Use Case – diagram uživatelských rolí

V rámci definic funkcionality webového portálu bylo pomocí vytvořeného Use Case diagramu definováno sedm hlavních uživatelských rolí včetně jejich možností interakce s portálem.



Obrázek 4: Use case diagram portálu Reklamář.cz

Zdroj: vlastní

Dle vytvořeného diagramu (Obrázek 3) budou v rámci portálu definovány tyto role:

- **Uživatel modulu dokladů a reklamací** – tento uživatel bude mít po nutné registraci možnost zadávat informace o svých nákupech (vložit daňový doklad), zahájit proces reklamace (založit reklamaci) a spravovat své vložené údaje.

- **Uživatel modulu hodnocení firem** – cílem tohoto uživatele je získat povědomí o zkušenostech s reklamacemi od ostatních uživatelů. Současně má také (po nucené registraci) možnost vkládat k libovolným firmám svá vlastní hodnocení. V rámci pilotního provozu bude možné přidávat hodnocení libovolně. V pozdější fázi, kdy již bude databáze hodnocení obsáhlejší, bude povolené jen hodnocení firem, u kterých uživatel zrealizoval reklamaci.
- **Uživatel modulu poradny** – uživatel hledající pomoc v rámci vytvořené poradny. Tento uživatel bude mít možnost volného prohlížení kompletního obsahu poradny, který bude strukturován do sekcí. V rámci celé poradny bude též možné vyhledávat připraveným nástrojem. Nadstandardní funkcí bude možnost zaslat přes kontaktní formuláře otázku poradci. Ten se následně s uživatelem spojí a pokusí se jeho dotaz co nejlépe zodpovědět.

Z hlediska správy portálu a přístupu do administračního rozhraní byly definovány tyto role:

- **Administrátor** se stará o kompletní správu účtů (administrátorských i uživatelských). Má možnost jednotlivé účty vytvářet, editovat, mazat, případně i blokovat. Povinností administrátora je ověřování zadaných hodnocení firem od uživatelů (tato povinnost bude fungovat jako ochrana před zneužitím tohoto nástroje).
- **Obchodník** je uživatel zodpovědný za propagaci webového portálu, který se stará o tvorbu a umístění bannerových kampaní, o úpravu pozic v žebříčku firem (na základě finančně zvýhodněných pozic), o práci se statistikami (na jejichž základě se upravují marketingové kampaně). V neposlední řadě obchodník na základě požadavků firem zadává uživateli v roli Copywriter tvorbu placených článků.
- **Copywriter** je zodpovědný za všechny texty na webu. Tyto texty vytváří tak, aby korespondovali s celkovým zaměřením portálu a také aby byly optimalizovány pro webové vyhledávače. Součástí toho je správa profilů firem, kde se stará o aktuálnost používaných log a popisů firem. Také se na základě požadavků od administrátora s rolí poradce stará o obsah a strukturu celé poradny.

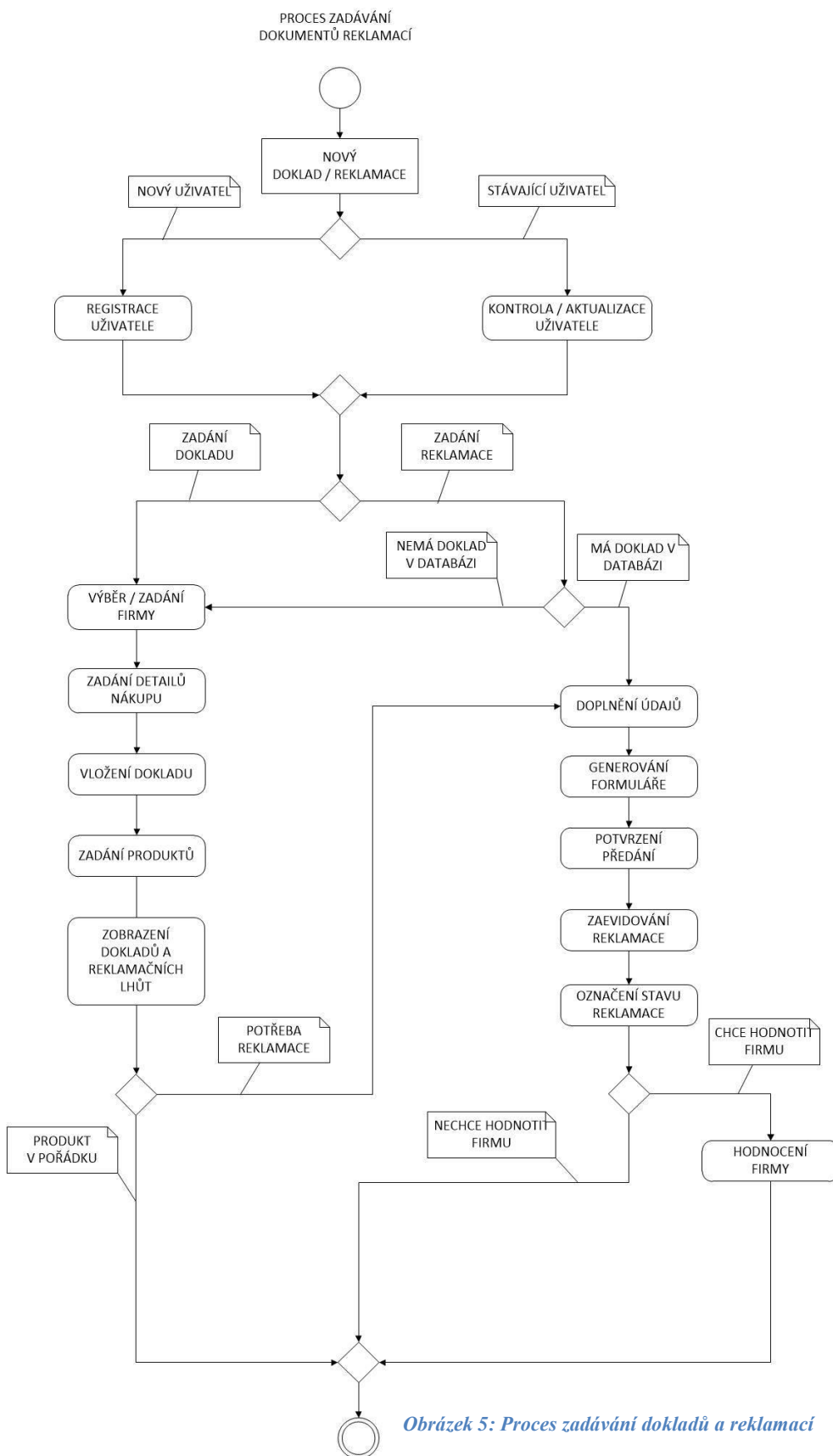
- **Poradce** zajišťuje sběr podkladů pro poradnu a také garantuje, že popsané postupy a rady jsou v souladu s aktuálně platnou legislativou. Současně také odpovídá na dotazy vznesené uživateli poradny.

2.3.2 Diagram aktivit – proces zadávání dokumentů a reklamací

Pro zpřehlednění celého procesu vkládání dokladu a zahájení případné reklamace byl vytvořen diagram aktivit (Obrázek 4). Uživatel přichází na portál s tím, že chce vložit daňový doklad či zahájit reklamaci. Pokud je na portálu poprvé, je nutné, aby se zaregistroval. V případě existujícího účtu je nutné, aby se do systému přihlásil. V případě přihlášení uživatele přes facebookový účet jsou zároveň aktualizovány uživatelské údaje. U přihlášení jsou evidovány též následující údaje: datum, čas a IP adresa, ze které se uživatel přihlašuje. Nyní záleží na uživateli, zda chce vložit daňový doklad či zahájit reklamaci.

- a) Při vkládání dokladu uživatel zvolí firmu z nabízeného seznamu, případně zadá firmu novou. Poté vyplní detaily nákupu (datum, poznámka), vloží soubor s dokladem (obrázek, PDF či jiný typ dokumentu) a zadá produkty, u nichž vyplní potřebné údaje (název položky, počet kusů, počet měsíců záruky a případně i poznámku). Po úspěšném založení dokladu má uživatel možnost procházet všechny své vložené doklady, upravovat jejich detaily a hlavně sledovat lhůty, v jejímž rámci jsou produkty reklamovatelné.
- b) Při zahájení reklamace mohou nastat dvě situace: uživatel již má vložený produkt v systému nebo reklamuje produkt, který zatím v databázi není. V druhém případě je vyžadováno zadání produktu do systému popsané v bodě a). Jakmile je v systému produkt založen a uživatel označil, že jej hodlá reklamovat, systém ho přesměruje na formulář, kde doplní nutné údaje (datum zahájení reklamace, způsob předání, popis vady, obsah balení apod.). Na základě těchto údajů je vygenerován reklamační formulář (dle platné legislativy). S tímto formulářem musí uživatel reklamovaný produkt předat na sběrné místo firmy, u níž produkt zakoupil. Systém v této fázi čeká, než uživatel potvrdí předání k reklamaci. Od této chvíle je v systému reklamace označená jako probíhající a systém začne automaticky sledovat zákonem dané lhůty pro reklamaci. Na konci tohoto procesu je od uživatele vyžadováno, aby poskytl zpětnou vazbu a označil, zda reklamace

byla úspěšná, neúspěšná a případně, zda se vyskytl nějaký problém. Po tomto kroku má uživatel možnost firmu ohodnotit.



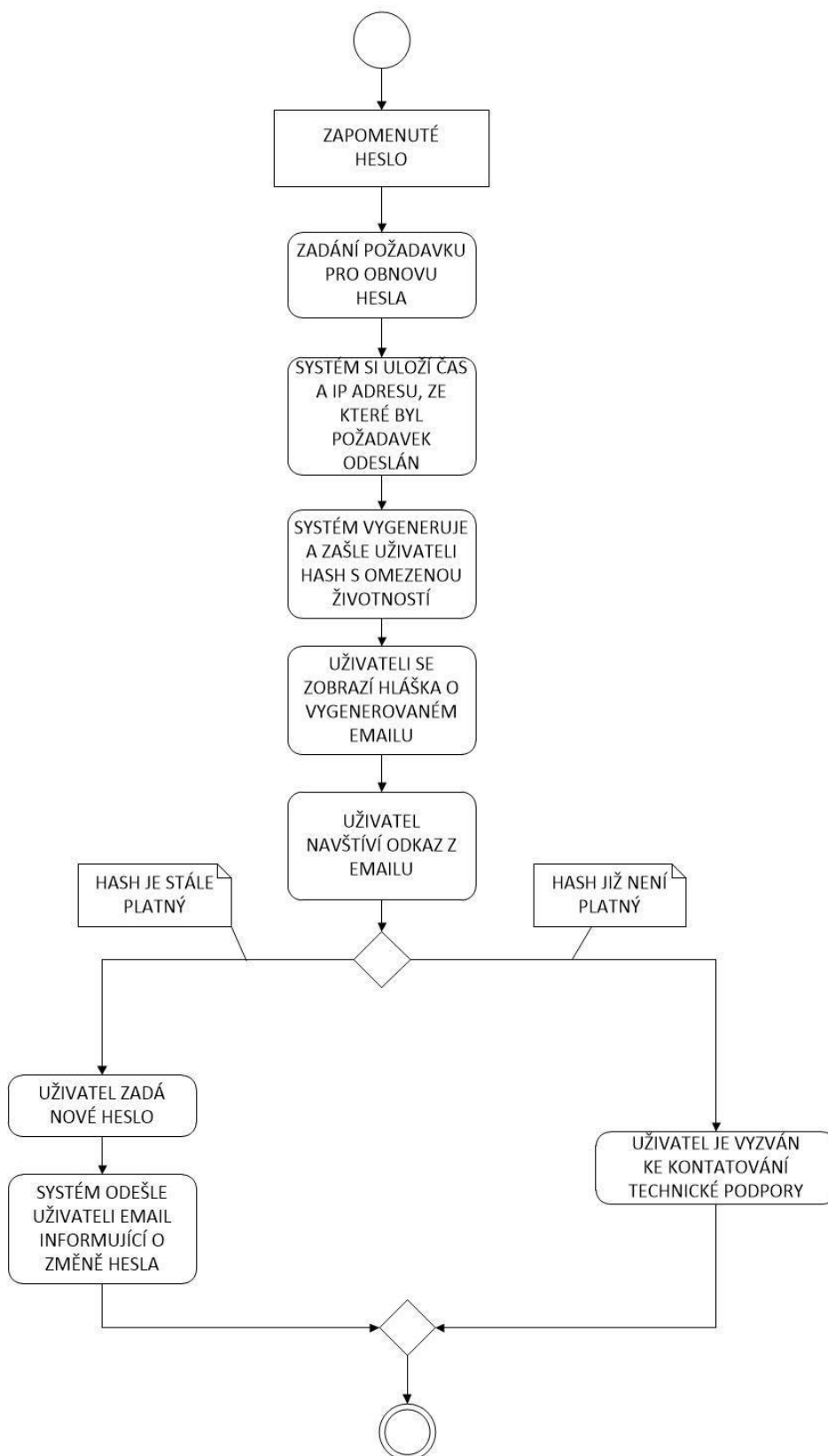
Obrázek 5: Proces zadávání dokladů a reklamací

Zdroj: vlastní

2.3.3 Diagram aktivit – proces obnovy hesla

Aby bylo splněno kritérium bezpečnosti, bylo rozhodnuto, že proces obnovy hesla, který bývá slabým místem webových portálů (kapitola 1.5.1), bude naddimenzován. Tento proces je zachycen v diagramu na obrázku 5. Výchozím bodem diagramu je moment, kdy si uživatel (registrovaný do aplikace klasickým způsobem pomocí svého emailu a ne pomocí sociální sítě – ten řeší zapomenutí hesla pomocí Facebooku) nemůže vzpomenout na své heslo a při pokusech o zadání mu systém zobrazuje varovnou hlášku o nesprávnosti přihlašovacích údajů. V této chvíli zadá požadavek do systému pro obnovu hesla. Systém si uloží do databáze časovou značku a IP adresu, ze které byl požadavek zaslán a následně vygeneruje a emailem zašle uživateli unikátní hash s omezenou životností (v řádu několika málo desítek minut). Součástí tohoto emailu je informace o vzniklém požadavku, přesném čase, kdy požadavek vznikl, a odkaz s vygenerovaným hashem. Uživateli je na webové stránce zobrazena hláška o odeslání ověřovacího emailu. Nyní uživatel musí přejít do své emailové schránky a tento odkaz navštívit. Pokud to nestihne v omezené době platnosti, systém mu zobrazí varovnou hlášku a doporučí mu kontaktování technické podpory, která mu heslo po ověření identity změní manuálně. Navštíví-li stránku v době, kdy je odkaz stále platný, je mu umožněno zadání nového hesla. Po zadání obdrží již jen informační email o tom, že heslo bylo změněno, s informací o čase změny hesla a IP adresy, ze které byla změna provedena. Veškeré údaje jsou uloženy v databázovém logu a uživatel je tak průběžně informován.[30]

PROCES OBNOVY HESLA



Obrázek 6: Proces obnovy hesla

Zdroj: vlastní

2.4 Specifika projektu

Krom předchozí definované funkcionality bylo nutné určit další klíčová specifika projektu, jež budou následně při vývoji zohledněna.

Portál bude vytvářen dle specifík pro moderní webový portál definovaných v první kapitole. Současně by měl obsahovat vizuálně zajímavou grafiku a vlastní designovou identitu. Pro dosažení dobrých pozic v rámci výsledků ve vyhledávačích je nutné provést během tvorby optimalizaci a vývoj nástroje pro úpravu jednotlivých parametrů sledovaných webovými vyhledávači.

Součástí aplikace by měl být automatizovaný skript spouštěný pomocí Cronu (nástroje pro automatické spouštění skriptů na serveru) pravidelně v nočních hodinách a odesílající automatické dva typy notifikací uživatelům. První typ notifikací bude informovat uživatele o tom, že jejich zadané produkty dosáhly nastaveného časového intervalu pro varování před expirací období reklamace produktů. Druhý typ notifikací bude informovat uživatele reklamující produkt o případném blížícím se termínu vyřízení reklamace.

Dalším specifikem řešeným v rámci portálu je vytváření profilu firmy při zakládání dokladu – pokud firma, u které uživatel zakoupil produkt, není v databázi portálu, je nutné zadat ji manuálně. Aby uživatel nemusel zdlouhavě vyplňovat celý formulář s údaji o firmě, bude připravena funkce pro načtení dat z obchodního rejstříku dle zadaného IČ. Toto načítání bude realizováno pomocí XML služeb portálu ARES (administrativní registr ekonomických subjektů).³ Toto řešení má však jednu nevýhodu a tou je limitovaný počet přístupů. Mezi 8:00 – 18:00 je možné zaslat 1000 dotazů na informace o firmách, mezi 18:00 – 8:00 následujícího dne lze zaslat 5000 dotazů. Při překročení tohoto limitu hrozí umístění webového portálu na blacklist. Ochranou před tímto potenciálním problémem bude vytvoření automatického čítače v aplikaci, který při překročení limitu možnost dynamického načítání z ARESu zablokuje a uživateli zobrazí pouze omluvnou zprávu.

Z hlediska bezpečnosti v rámci portálu je zahrnuto i šifrování souborů. Pro zajištění bezpečnosti a získání důvěry uživatelů, kteří budou do systému nahrávat své citlivé informace (účetky, faktury), dojde k šifrování veškerých nahrávaných souborů na server.

³ Vzorový příklad poskytovaného XML je dostupný na adrese: http://www.info.mfcr.cz/cgi-bin/ares/darv_std.cgi?ico=27074358.

Při nahrávání souboru bude pro každý soubor a každého uživatele vytvořen unikátní hash. Pomocí tohoto hashe bude celý obsah souboru zašifrován a uložen na webovém úložišti s vygenerovaným názvem a unikátní koncovkou cCf. Informace o souboru se následně uloží do databáze. Pokud uživatel bude chtít zobrazit či stáhnout soubor, dojde k zaslání požadavku na serverový skript, který dle identifikátoru nalezne požadovaný soubor na serverovém úložišti, sestaví šifrovací hash a s jeho pomocí rozšifruje soubor. Tento soubor bude s původním názvem (načteným z databáze) odeslán do uživatelova prohlížeče. Uživatel tak uvidí soubor ve stejné podobě, v jaké ho nahrál. Výhodou tohoto řešení je, že provozovatel nebude mít k nahraným souborům přístup a v případě napadnutí webu najde případný útočník pouze složky se „zvláštními“ soubory, které nebude možné otevřít v žádném programu.

Pro další zlepšení zabezpečení bude celý portál využívat HTTPS, čím se zabrání odposlechnutí nešifrované komunikace mezi uživatelem a serverem. Uživatelská hesla budou ukládána pomocí šifry BCRYPT a minimální délka vyžadovaného hesla bude 8 znaků.

Pro plánovanou mobilní aplikaci dojde i vývoji API, se kterým bude aplikace na základě zabezpečených protokolů komunikovat (nahrávat a stahovat data).

2.5 Výběr technologie

Na základě předchozích specifikací bylo možné rozhodnout, na jakých technologiích bude portál postaven.

Pro psaní hlavních zdrojových kódů bylo zvoleno PHP ve verzi minimálně 5.4 a jako databáze MySQL, v rámci kterého budou krom tabulek vytvářeny též pohledy a procedury. Komunikace s databází bude v rámci PHP probíhat pomocí knihovny PDO. Pro doplnění možností PHP o šifrování dat pomocí BCRYPTU bude využita knihovna Password compat. Pro šifrování souborů dojde k použití knihovny PHPSecLib, která doplňuje PHP o další šifrovací algoritmy.

Pro zlepšení uživatelské interakce bude využit JavaScript a zejména jeho nadstavba jQuery ve verzi 1.11. Současně s ní budou také využity prvky jQuery UI pro automatické našeptávače. Pro zobrazení tabulek a práci s nimi bude implementována knihovna DataTables. Pro možnost hodnocení firem pomocí hvězdiček dojde k využití knihovny

RateIt. V administrátorském prostředí budou DataTables doplněny o knihovnu Buttons, která umožní tabulky exportovat do různých formátů (PDF, CSV apod.). Aby byla úvodní stránka zajímavější, bude na ní umístěn takzvaný slider Slideme2, což je nástroj prezentující sérii obrázků a popisků ve smyčce. U administračního nástroje budou tyto knihovny ještě doplněny o WYSIWYG editor CKeditor a jeho rozšíření KCFinder pro umožnění nahrávání obrázků a jiných různých souborů do webových stránek.

2.6 Podklady pro výběr vhodného webhostingu

Dle zvolených technologií byly určeny klíčové parametry pro zakoupení vhodného webhostingu a registraci domény. Požadované parametry pro webhosting byly:

- PHP v minimální verzi 5.4
- Databáze MySQL (výhodou je nainstalovaný nástroj pro správu PHPMyAdmin)
- Velké datové uložení s možností případného rozšíření kapacity
- Integrovaný Cron
- Podpora velkého množství zobrazení webu najednou
- Automatické zálohování (alespoň 1x týdně)
- Podpora HTTPS
- Dobré reference o provozovateli webhostingu a spolehlivá technická podpora
- Přijatelná cena

Na trhu je velké množství provozovatelů webhostingu, kteří tato kritéria splňují, nakonec bylo rozhodnuto o volbě firmy Wedos. S touto firmou mají autoři portálu Reklamář.cz velmi dobré zkušenosti. Pro zahájení provozu byla zvolena jedna ze základních variant nabízených webhostingů s tím, že stane-li se projekt v prostředí českého internetu populárním a tato varianta již nebude dostačovat, bude možné rychle přejít na výkonnější virtuální server s lepšími parametry. Aby mohla začít práce grafiků a nebylo případně nutné měnit název vznikajícího portálu, byla u firmy Wedos zaregistrována doména Reklamář.cz a současně zakoupen webhosting.

3 Tvorba moderní webové aplikace

Po dokončení analýz vznikajícího webového portálu Reklamář.cz a sestavení kompletních požadavků v předchozích kapitolách bylo možné zahájit fázi vývoje.

3.1 Databázový model (fáze návrhu)

Prvním krokem bylo vytvoření kompletního databázového modelu tak, aby odpovídal definovaným požadavkům vývoje a aby splňoval kritéria pro moderní webový portál.

Pro práci s databází byly vytvořeny tři přístupové účty:

- **účet pro kompletní správu databáze** – přístupové údaje k tomuto účtu nebudou součástí žádných zdrojových kódů, v případě potenciálního útoku útočník nebude mít možnost tento účet zneužít ke změně databázové struktury;
- **účet pro webový portál (UW)** – tento účet má možnost pouze dotazování se do vybraných tabulek a spouštění vybraných procedur;
- **účet pro administraci webového portálu (UA)** – oproti účtu pro webový portál je tomuto účtu umožněno zobrazení všech tabulek a spouštění všech procedur.

Princip rozdělení účtu na UW a UA spočívá v tom, že uživatelský účet bude využíván širokou škálou uživatelů a není tak nutné, aby měl přístup ke všem funkcím, které databáze poskytuje. Tím je zajištěno vyšší zabezpečení databáze.



Obrázek 7: Databáze portálu Reklamář.cz

Zdroj: vlastní

Poté bylo možné vytvořit v databázi následující tabulky, zobrazené na obrázku 7:

- **tblaccess** – tabulka pro UA, ve které jsou uchovávány přístupové údaje administrátorů a jejich uživatelské role.
- **tbldoklady** – hlavní tabulka pro správu dokladů, kde má každý doklad svůj identifikátor (`id_dokladu`) a je propojen s tabulkami uživatelů (`tbluser`) a firem (`tblfirem`). V tabulce je zároveň uložen datum nákupu, datum vložení a parametr `vis`, který určuje, zda je vložený doklad viditelný.
- **tbldoklady_polozky** – tabulka pro přiřazení zakoupených produktů k dokladům. Každá položka je propojena s tabulkou stavů (`tblstavu`). Ukládány jsou údaje: název položky, počet zakoupených kusů, počet měsíců záruky (kde 0 určuje doživotní záruku a datový typ NULL žádnou záruku – to se týká například nákupu použitých produktů), termínu konce záruky, volitelné poznámky (aby si uživatel mohl spravovat upřesňující informace, např. „Přehrávač v kuchyni“) a parametr `vis`.
- **tbldoklady_soubory** – tabulka pro správu nahraných dokumentů, v níž může být ke každému dokladu (`id_dokladu`) přiřazeno více souborů. U souborů se ukládá jejich původní název a časová značka „`timest`“ určující, kdy byl soubor nahrán.
- **tblfirmy** – zde jsou uloženy všechny firmy, se kterými portál pracuje. U firem je uložen jejich název, IČ, odkaz na webové stránky a adresa (ulice, město, PSČ a stát). Pro možnost zvýhodnění v rámci žebříčku hodnocení je firma propojena s tabulkou propagace (pomocí `id_propagace`).
- **tblfirmy_hodnoceni** – klíčová tabulka pro hodnocení firem. Každé hodnocení je přiřazeno k firmě (`tblfirmy`) a uživateli (`tbluser`). U každého hodnocení bude uloženo jméno (či přezdívka), jaké si hodnotitel zvolil, parametr o zobrazení jména (pokud nebude chtít zobrazit své jméno, u hodnocení bude zobrazeno jen první a poslední písmeno jména, přičemž výsledný formát bude doplněn hvězdičkami. Tzn., například u jména Petr Novák bude zobrazeno ‘P***K’), hodnocení kladů (plusy) a záporů (minusy) reklamace a případný komentář. Dále bude uložena časová značka a číselné hodnocení (hvězdičkový systém). Posledním parametrem je sloupec „`schváleno`“, který určuje schválení hodnocení administrátorem.
- **tblfirmy_kategorie** – tabulka pro rozřazení firem do kategorií. U každé kategorie je specifikován název, adresa URL pro prohlížeč (`link_kategorie`), pořadí zobrazení

kategorií (aby administrátoři mohli případně určovat pořadí dle popularity) a parametr vis.

- **tbllog** – interní tabulka pro evidování veškerých činností s daty v databázi. Přístup do této tabulky má pouze UA (který má práva pouze na výběr). Struktura této tabulky je: id, datum vložení záznamu (idate), tabulka, u které ke změně došlo (location), vykonávaná akce (action), uživatel (user), klíč záznamu v tabulce (table_id), upravená data (table_data) a IP adresa, ze které změna proběhla.
- **tblpropagace** – v této tabulce jsou definovány jednotlivé úrovně pro propagaci firem. Každá úroveň má své id, název a číselné vyjádření úrovně, které bude následně použito při výpisu firem v žebříčcích.
- **tblreklamace** – hlavní tabulka pro reklamace, kde má každá reklamace následující sloupce: id reklamace, id dokladu (nad kterým je reklamace realizována), vazba na způsob předání a požadovaný způsob vyřízení reklamace (id_reklamace_predani a id_reklamace_vyrizeni), datum zahájení reklamace, datum předání k reklamaci a datum ukončení reklamace, doplňující informace ke zboží, popis vady, obsah předaného balení, ostatní informace a požadavky, textový popis jiného vyřízení (v případě že při výběru požadovaného způsobu vyřízení zvolí „jiný“), interní identifikátor o uskutečněném hodnocení a poznámku.
- **tblreklamace_historie** – tabulka pro uložení průběhu reklamace, kde je uloženo, kdy u jaké reklamace nastal jaký stav.
- **tblreklamace_polozky** – tabulka pro propojení vložených produktů a reklamací. Dodatečným parametrem je zde počet předchozích reklamací, který je buď automaticky dopočítán, nebo je možné nastavit jej manuálně.
- **tblreklamace_predani** – pomocná tabulka pro volbu způsobu, jakým uživatel předal produkt k reklamaci (např. „poštou“, „kurýrem“, „osobní předání na pobočce“).
- **tblreklamace_vyrizeni** – pomocná tabulka pro volbu způsobu, jakým bude reklamace vyřízena (např. „oprava zboží“, „výměna zboží“).
- **tblsablony_mailu** – zde jsou uloženy vzory šablon, které se odesílají emailem. Výhoda uložení šablon v databázi spočívá v možnosti dynamicky je upravovat a přizpůsobovat je aktuálním trendům. Každá šablona má textové id, s jehož pomocí je vybrána v rámci zdrojového kódu, předmět emailu a odesílaný text zprávy.

- **tblstatistiky** – tabulka sloužící jako zdroj dat pro interní statistiky portálu. U každého záznamu je určeno id, kategorie (id_stat_kategorie), IP adresa a datum vložení.
- **tblstatistiky_kategorie** – zde jsou specifikovány jednotlivé typy sledovaných statistik. Každá statistika má své id, název a popis. Sledovaným statistikami jsou například počet přístupů k portálu ARES, počet zobrazení vybraných článků apod.
- **tblstav** – pomocná tabulka pro určení stavu produktů. Uložené stavy jsou například: „založeno“, „blíží se konec reklamační lhůty“, „v procesu reklamace“ apod.
- **tblstav_reklamace** – pomocná tabulka pro určení stavu, v jakém se reklamace nachází.
- **tbluser** – tabulka s kompletními údaji o uživateli. Každý uživatel má své id (id_user), jméno, příjmení, email, hash s heslem, případné facebookové jméno, adresu, město, stát, PSČ, datum a IP posledního přihlášení a parametr vis.
- **tbluser_notifikace** – tabulka sloužící ke konfiguraci počtu dní pro odesílání automatických notifikací. Při zakládání uživatele jsou zde automaticky nastaveny výchozí hodnoty, které uživatel může následně změnit v sekci „Můj účet“.
- **tbluser_hash** – pomocná tabulka pro ukládání generovaných hashů v procesu obnovy hesla. Každému hashi je přiřazená časová značka, pomocí níž je následně v aplikaci vypočítána doba platnosti.
- **tblwiki** – hlavní tabulka poradny, v níž jsou uloženy jednotlivé články. Každý článek zařazený v kategorii (id_kategorie) má položenou otázku (např. „Jak reklamovat“), krátkou a dlouhou odpověď, parametr link pro adresu a tagy, které budou sloužit pro vyhledávače.
- **tblwiki_kategorie** – poslední tabulka slouží pro specifikování názvů kategorií, jejich adres a tagů pro vyhledávače.

Pro rozšíření funkčnosti byla databáze doplněna o:

- **vi_souhrn_uzivatele** – databázový pohled pro zjištění statistik o uživateli. Sledovanými statistikami jsou počty: vložených dokladů a produktů, realizovaných reklamací a hodnocení firem.

- **fn_statistiky** – funkce pro rychlé zjištění hodnoty zvolené statistiky. Parametry této funkce jsou `id_statistiky` a časový rozsah, v rámci jakého má být statistika interpretována.

Pro splnění kritérií zabezpečení moderního webového portálu nejsou ve zdrojovém kódu obsaženy přímé příkazy pro práci s daty (INSERT, UPDATE, DELETE). Z toho důvodu je pro každou tabulku vytvořena obslužná procedura, která na základě zadaných parametrů předpřipravených instrukcí provádí práci s daty.

```
CREATE DEFINER='root'@'localhost' PROCEDURE `sp_doklady_polozky`(
  IN i_typ VARCHAR(10),
  IN i_id_polozky INT,
  IN i_id_dokladu INT,
  IN i_id_stav INT,
  IN i_nazev_polozky VARCHAR(150),
  IN i_pocet DOUBLE,
  IN i_zaruka_mesicu INT,
  IN i_datum_ukonceni DATETIME,
  IN i_poznamka TEXT,
  IN i_email VARCHAR(100),
  IN i_ip VARCHAR(255)
)
BEGIN
  CALL sp_log('tbldoklady_polozky',i_typ,i_email,i_id_polozky,concat(i_id_dokladu,' ',i_id_stav),i_ip);

  IF (i_typ = 'INSERT') THEN
    INSERT INTO tbldoklady_polozky
      (id_stav,id_dokladu,nazev_polozky,pocet,zaruka_mesicu,datum_ukonceni,poznamka,vis)
    VALUES
      (i_id_stav,i_id_dokladu,i_nazev_polozky,i_pocet,i_zaruka_mesicu,i_datum_ukonceni,i_poznamka,1);

  ELSEIF (i_typ = 'UPDATE') THEN
    UPDATE tbldoklady_polozky SET nazev_polozky = i_nazev_polozky, zaruka_mesicu = i_zaruka_mesicu,
      poznamka = i_poznamka,pocet = i_pocet
    WHERE id_polozky = i_id_polozky;

  ELSEIF (i_typ = 'STAV') THEN
    UPDATE tbldoklady_polozky SET id_stav = i_id_stav WHERE id_polozky = i_id_polozky;

  ELSEIF (i_typ = 'DELETE') THEN
    UPDATE tbldoklady_polozky SET vis = 0 WHERE id_polozky = i_id_polozky;

  ELSEIF (i_typ = 'RENEW') THEN
    UPDATE tbldoklady_polozky SET vis = 1 WHERE id_polozky = i_id_polozky;

  END IF;
END
```

Obrázek 8: Procedura `sp_doklady_polozky`

Zdroj: vlastní

Vzorovým příkladem použitých procedur je procedura `sp_doklady_polozky` zobrazená na obrázku 8. Tato procedura slouží k obsluze tabulky `tbldoklady_polozky`. Pro rozhodnutí o prováděné akci slouží parametr `i_typ`. Díky tomuto parametru procedura dokáže rozhodnout, zda bude zadaná data vkládat (`i_typ == 'INSERT'`), upravovat (`i_typ == 'UPDATE'`), nastavovat jejich stav (`i_typ == 'STAV'`) či upravovat jejich viditelnost

(`i_typ == 'DELETE'` a `i_typ == 'RENEW'`). Před zahájením každé operace s daty je zavolána procedura `sp_log`, která slouží pro ukládání veškerých činností s tabulkami do tabulky `tbllog`.

Po dokončení tvorby databáze a jejím následném ověření bylo možné zahájit druhou část fáze návrhu – fázi vývoje.

3.2 Vývoj moderního webového portálu (fáze návrhu)

Cílem této fáze bylo vyvinout nad vytvořenou databází webový portál, který bude splňovat specifika moderního webového portálu (a to včetně podpory mobilních zařízení). Jako první bylo vyvinuto jádro samotného portálu, jehož primárním úkolem je komunikace s databází pomocí předpřipravených parametrizovaných dotazů (ať už se jedná o obsluhu databázových procedur, či dotazů na data). Postupně byly k jádru doplněny všechny požadované knihovny (Password compat, PHPSecLib a rozhraní pro sociální síť Facebook). Také došlo k vývoji a otestování napojení na ARES a integrovaného nástroje pro šifrování uživateli nahrávaných souborů.

Specifickou částí jádra je využití transakce při vkládání nového dokladu – transakce je zahájena při vložení dokladu do tabulky `tbldoklady` a ukončena po dokončení vkládání produktů do `tbldoklady_soubory`. Pokud se v některé části tohoto procesu vyskytne chyba, je pomocí transakce možné všechny předcházející kroky zrušit a uživateli pouze vypsat chybové hlášení s prosbou o opětovné vložení dokladu.

Pro zlepšení navigace na portálu, a následně i poskytnutí lepších prostředků pro sledování provozu na webových stránkách, byly v rámci vývojového serveru upraveny pravidla konfiguračního souboru „`htaccess`“. Tato úprava posloužila k tomu, že namísto klasické adresy URL „`www.reklamar.cz/index.php?cil=Poradna`“ bude výsledná URL vypadat „`www.reklamar.cz/Poradna`“. Jako parametr cíl tak slouží první text za lomítkem.

Při následném návrhu šablony byly vzaty v potaz požadavky na vzhled a na ovládání moderního portálu. Šablona ve velké míře využívá vlastností HTML5 a CSS3, přičemž v rámci HTML5 využívá: nové tagy pro definování jednotlivých prvků webové stránky (header, nav, section apod.), nový tvar hlaviček, rozšiřující prvky pro formuláře (parametr `required` či nové typy formulářových polí typu „`number`“ a „`email`“) a také microdata pro roboty indexující weby pro webové vyhledávače. V rámci CSS3 šablona využívá

zejména: nové selektory pro stylování dle nastavitelných pravidel, vlastnost Box-sizing pro úpravu zobrazení jednotlivých prvků webové stránky, media-queries pro zajištění podpory mobilních zařízení responsivním zobrazením, v neposlední řadě také stíny u vybraných textů a prvků. Pro zatraktivnění grafického vzhledu webu nejsou použity klasické fonty, ale fonty poskytované portálem fonts.google.com – v našem případě jeden z nejpopulárnějších fontů, Open Sans. Výhodou těchto fontů je moderní vzhled a stejné zobrazení na všech platformách. [31]

Šablona byla nejprve vytvořena jako hrubý koncept (Příloha A 1), který byl po konzultacích vývojového týmu s externím grafickým poradcem upraven do finální podoby (Příloha A 2). Výsledný vzhled tedy obsahuje:

- **Hlavičku** – obsahující logo (které zároveň funguje jako odkaz na domovskou stránku), slogan webu (který bude následně během roku copywriter upravovat tak, aby reagoval na aktuální dění), malý vyhledávací formulář a uživatelský blok, v němž je zobrazeno tlačítko pro přihlášení nepřihlášeného uživatele či jméno přihlášeného uživatele s integrovaným rozbalovacím menu.
- **Plovoucí menu** – pokud uživatel začne rolovat stránkou, menu opustí svou pevně danou pozici a zůstane vždy na vrcholu zobrazené části stránky. Díky tomuto řešení uživatel stále ví, na jaké stránce se nachází, a má tak možnost rychlé navigace v rámci celého portálu. Součástí toho menu jsou odkazy na jednotlivé části portálu (Úvod, Reklamace, Hodnocení firem, Poradna a Kontakt).
- **Sekci pro obsah stránek** – zde je šablona specifická tím, že má již předpřipravené bloky pro uspořádání stránky na šířku dle následujících schémat, kde číslo v závorce určuje procentuální využití šířky stránky:
 - 1 blok (100%)
 - 2 bloky (50% a 50%)
 - 3 bloky (50%, 25% a 25%)
 - 4 bloky (každý o šířce 25%)
- **Patičku** – zahrnující informace o portálu, zjednodušenou navigaci, blok s odkazy na sociální sítě (v jehož rámci budou v budoucnu i zobrazeny odkazy na stažení mobilních aplikací) a krátký text „O nás“ vysvětlující vznik a hlavní myšlenku portálu.

Z hlediska podpory mobilních zařízení (u nichž hraje velkou roli individuální rozlišení obrazovek) dochází při zmenšování rozlišení k těmto krokům:

1. Jako první je při nedostatku místa na stránce skryt slogan (Příloha B 1). To z důvodu, aby zbyl prostor pro vyhledávací formulář a uživatelský blok.
2. Při překročení hranice 800px na šířku dochází k zásadní změně – z hlavičky mizí vyhledávací formulář a uživatelský blok. Také mizí plovoucí menu a bloky v patičce a sekci pro obsah stránky jsou přeskupeny pod sebe. Nově je zobrazena ikona pro rozbalovací menu, do kterého se přesunul vyhledávací formulář, kompletní menu a uživatelský blok. (Příloha B 2)

Tím je vyřešeno responsivní zobrazení, uživatel mobilní verze přitom díky kompaktnímu zobrazení není ochuzen o žádnou z poskytovaných funkcí.

Pro rozšíření webového portálu o další možnosti interakce byly použity zvolené JavaScriptové knihovny, díky nimž bylo možné vytvořit vlastní funkce. Jednou z použitých funkcí jsou například automatické našeptávače. Pokud uživatel využívá vyhledávání (či vybírá firmu), jsou mu pod vyhledávacím formulářem dynamicky zobrazeny výsledky (načítané v reálném čase pomocí AJAXu).

Nad hotovou klientskou částí webového portálu byla následně zahájena implementace redakčního systému (administrace). V tomto případě byla implementace výhodnější než vývoj zcela nového nástroje – web využívá nástroj cAdmin v. 1.0, který vyvinul autor této práce již dříve a používá jej v rámci několika dalších provozovaných webových portálů. Díky pozitivní zpětné vazbě bylo možné jej rychle a efektivně nasadit i v tomto případě.

V poslední části vývoje zbývalo vyvinout API pro plánovanou mobilní aplikaci. Součástí tohoto API je rozhraní pro ověřování a vytváření uživatelů, přenos uživatelských dat, hodnocení firem a znalostní báze. Aby nemohlo dojít ke zneužití API, každý požadavek musí mít v sobě zahrnut jedinečný identifikátor.

Před ukončením vývoje bylo ještě nutné veškeré části vyvinutého portálu otestovat na lokálním serveru. V rámci testování a doladování došlo k sepsání dokumentace a definic budoucích testů.

3.3 Reálná implementace do testovacího prostředí

Protože byl vývoj již dokončen a portál připraven k nahrání do testovacího prostředí, bylo možné zahájit konfiguraci webhostingu. Konfigurace byla zahájena vytvořením databázového prostoru a uživatelských účtů databáze. Následně byly vytvořeny a nakonfigurovány vybrané emaily (např. „info@reklamar.cz“). Do databázového prostoru byla nahrána kompletní databázová struktura, očištěná od testovacích dat. Současně byla databáze naplněna administrátorskými účty a testovacími texty (profily firem, články poradny). Také byl v rámci domény nainstalován ověřený HTTPS certifikát.

Protože bylo nutné provést důkladné testování a nebylo by moudré nechávat portál plně přístupný (jak pro případné budoucí uživatele, tak zejména pro automatické roboty indexující webové stránky pro webové vyhledávače), byly provedeny dvě zásadní úpravy. Konfigurační soubor .htaccess byl doplněn o podmínku, která při přístupu na doménu reklamar.cz automaticky zobrazí výchozí stránku od webhostingu s informací o aktivní službě. Druhou úpravou, zaměřenou na roboty, bylo doplnění hlavičky portálu o meta atribut „ROBOTS“ s parametrem „NOINDEX, NOFOLLOW“, který těmto robotům řekne, že v současné fázi nesmí tento portál indexovat a ani sledovat zrealizované změny.

Po zprovoznění webového portálu na doméně a následné konfiguraci všech potřebných částí mohlo být zahájeno celkové testování.

3.4 Celkové testování

V této fázi byl portál předán k testování vybraným testovacím uživatelům. Podle vytvořené dokumentace došlo k provádění dvou základních typů testů:

- **Zátěžové a výkonnostní testy** – cílem těchto testů bylo ověřit rychlosti načítání jednotlivých stránek celého portálu a ovlivnění načítání stránek větším množstvím paralelně přistupujících uživatelů.
- **Testy uživatelské přívětivosti** – výstupem těchto testů bylo subjektivní hodnocení ergonomie a intuitivnosti používání celého portálu. Výhodou využití testerů, kteří se nepodíleli na vývoji, byl fakt, že na rozdíl od vývojového týmu viděli portál poprvé a mohli tak provádět nezájaté testy.

Při zátěžových testech byl použit online nástroj Loader.io, který po nutné registraci umožňuje simulace přístupu vybraného počtu uživatelů – např. v příloze C 1 je zobrazen záznam jednoho z testů, který simuloval zátěž webového portálu 1000 uživatelů v rámci jedné minuty. Výsledky těchto testů byly uspokojivé.

Dalším použitým nástrojem pro výkonnostní testy byl online portál webpagetest.org. Ten se zaměřuje hlavně na analýzu načítaných dat a hodnocení webových stránek dle vlastních metrik. Výstupem je sada doporučení pro případné optimalizace a úpravy – například při jednom z testování (příloha C 2) bylo zjištěno, že je zapotřebí zapracovat na kompresi obrázků (následná analýza odhalila problém v obrázcích slideru na úvodní stránce).

Aplikační logika a intuitivnost byla dle hodnocení testerů na výborné úrovni, avšak některé grafické prvky bylo potřeba upravit (došlo ke zvětšení tlačítek na přihlašovací formuláři a barevnému odlišení výsledků v rámci integrovaného vyhledávání).

Díky výsledné spokojenosti s testy bylo možné ukončit tuto fázi a přejít k zahájení provozu.

3.5 Přípravy k provozu (fáze zavádění systému)

Z databáze nyní bylo možné odstranit všechna testovací data a nahradit je daty reálnými. V uložišti nahrávaných souborů byly smazány testovací soubory. Ve zdrojových kódech a konfiguraci serveru bylo možné vypnout všechna pomocná hlášení a výpisy pro testování.

3.6 Zkušební období

V této fázi by mělo dojít plnému zahájení provozu webového portálu, nicméně v rámci vyvíjeného portálu Reklamář.cz bylo spuštění zrealizováno neveřejně. V době odevzdání této diplomové práce je portál funkční na skryté adrese a reálně ho využívají jen vybraní uživatelé potvrzení administrátory (samovolná registrace do portálu je zakázána).

Zahájení reálného provozu bylo původně plánované na půlku dubna 2016, nicméně z důvodu plného časového vytížení autorů portálu byl tento termín posunut na přelom července a srpna roku 2016. Aby nebyl portál nevyužit, bylo rozhodnuto o zpřístupnění vybraným uživatelům s cílem dodatečného ověření funkčnosti. Důvodem tohoto odložení byl také fakt, že díky unikátnosti tohoto projektu v prostředí českého internetu by mohlo dojít k jeho rychlému okopírování a tím ke vzniku (zatím neexistující) konkurence. Proto

je nezbytné, aby celé vývojové, provozní i marketingové týmy měly volné časové kapacity a dokázaly zajistit úspěšný start projektu včetně získání široké uživatelské komunity.

Při plánovaném veřejném startu bude celý provoz portálu v prvních dnech pečlivě monitorován. Cílem bude odhalit, jak se reální uživatelé v rámci portálu chovají a dle toho v první fázi upravit články na webu a marketingové kampaně. Marketingové kampaně budou zahájeny ihned po startu.

3.7 Rutinní provoz, údržba a případné optimalizace

V rámci běžného provozu jsou plánovány průběžné kontroly funkčnosti a vznikajícího obsahu. Vzhledem k unikátnosti celého projektu je možné, že celý portál bude rozšířen o nové funkce – pokud by k tomu došlo, budou znovu zopakovány kroky SDLC.

Výhodou navrženého řešení je snadná úprava obsahu a grafiky tak, aby nebylo potřeba provádět žádné zásadní změny ve zdrojovém kódu. Tím je umožněno reagovat velice rychle na změny ve společnosti a na stále se měnící trendy ve webové grafice i v tvorbě moderních webových portálů. V případě změny legislativy zabývající se nákupem a reklamami zboží a služeb je také možné díky navržené struktuře tyto změny rychle do funkcionality portálu zahrnout.

3.8 Výsledný portál

Vybrané sekce webového portálu Reklamář.cz, který vznikl v rámci praktické části této diplomové práce, budou popsány v následujících kapitolách.

3.8.1 Hlavní stránky

První stránka, kterou uživatel přistupující na webový portál Reklamář.cz uvidí, je stránka **Úvod** (příloha A.2). Cílem této stránky je uživatele zaujmout tak, aby na stránce nejen zůstal, ale aby se též zaregistroval a začal využívat poskytovanou funkcionalitu. Hlavním prvkem stránky je dynamicky se měnící slider zobrazující obrázky a klíčové fráze popisující portál (např. „Vyřešte reklamace jednoduše“ či „Ověřte si svá práva a povinnosti“). Součástí těchto popisků jsou i tlačítka pro rychlé vložení daňového dokladu a založení reklamace. Pod sliderem je v levém hlavním bloku umístěn popis portálu a nejnovější zadaná hodnocení uživatelů. V pravém bloku je zobrazen žebříček nejlépe

hodnocených firem v rámci hodnocení reklamací a také nejnavštěvovanějších článků poradny v posledním týdnu.

Druhou hlavní stránkou jsou **Reklamace**, které popisují celý reklamační proces a poskytují uživateli důvody proč pro reklamace využívat právě tento portál.

Následující stránkou je sekce **Hodnocení firem** (příloha D 1), která umožňuje rychle vyhledávat nad zadanými hodnoceními firem. Firmy jsou rozdělené do kategorií dle svého oboru podnikání. Na úvodní stránce uživatel vidí u každé firmy její logo, průměrné hodnocení (vyjádřené hvězdičkami). V levém menu jsou vypsané kategorie jednotlivých firem. Pokud si uživatel zobrazí detail firmy, uvidí rozepsaná hodnocení od jednotlivých hodnotitelů.

Další hlavní stránkou je sekce **Poradna** (příloha D 2), která funguje jako znalostní báze zabývající se otázkami ohledně nákupu produktů, reklamacemi a obecnými právy a povinnostmi zákazníka. Na úvodní stránce Poradny jsou zobrazeny nejčastější dotazy a jejich krátké odpovědi. V levém menu jsou kategorie Poradny (např. „Nákup“ či „Reklamace“). Pokud uživatel u otázky klikne na tlačítko „Zobrazit více informací“, je přesměrován na stránku s kompletní odpovědí.

Poslední z hlavních stránek je **Kontakt** (příloha D 3), kde mají uživatelé šanci získat kontaktní údaje na provozovatele webového portálu, technickou podporu a také Poradnu. Součástí stránky je i kontaktní formulář.

3.8.2 Klientská část

Pro vstup do klientské části webového portálu je vyžadováno, aby se uživatel zaregistroval nebo přihlásil (příloha D 4). Po úspěšném přihlášení je uživatel přesměrován na stránku, ze které se přihlašoval, a v uživatelském menu mu jsou zpřístupněny sekce: Mé doklady, Mé reklamace, Zadaná hodnocení, Nastavení a tlačítko pro odhlášení se z portálu. Pokud již nějaký z uživatelových produktů dosáhl jednoho ze sledovaných notifikačních termínů, je uživateli po přihlášení zobrazeno varování o tomto termínu.

V rámci sekce **Mé doklady** (příloha D 5) může uživatel přidávat nové doklady či spravovat doklady stávající. Při správě dokladů uživatel vidí všechny vložené produkty a jejich detaily (firma, u které produkt zakoupil, počet kusů, datum nákupu, vypočítaný

konec záruky, počet měsíců záruky, stav a poznámka). Každý z těchto produktů je možné odstranit či v rámci detailu upravit. Při vkládání dokladu (příloha D 6) uživatel zvolí (nebo manuálně zadá firmu), vybere datum nákupu, nahraje daňový doklad a nakonec vloží produkty. Zadané doklady se po vložení zobrazí v sekci Mé doklady.

Sekce **Mé reklamace** funguje obdobně jako Mé doklady s tím, že uživatel vidí jen produkty, které jsou právě v procesu reklamace nebo již reklamačním procesem prošly.

V rámci sekce **Má hodnocení** je uživateli umožněno hodnotit firmy pomocí formuláře či spravovat již zadaná hodnocení. Tato sekce uživateli poskytuje okamžitý přehled o zrealizovaných reklamacích a může si tak snadno připomenout, jak reklamace probíhala.

Poslední částí je stránka **Nastavení**, která je rozdělena na čtyři hlavní části:

- **Mé údaje** – zde má uživatel možnost správy svých údajů zadaných při registraci (včetně změny hesla). Také zde vidí informaci o svém posledním přihlášení.
- **Intervaly notifikací** – v této části si uživatel může nastavit počet dní před koncem reklamačního období a počet dní před vyřízením reklamace, v nichž mu bude zaslána automatická notifikace.
- **Funkce** – blok s rychlými odkazy pro zadání daňového dokladu, zahájení reklamace a vložení hodnocení firmy.
- **Statistiky** – blok se souhrnnými statistikami, kde uživatel přehled o své činnosti v rámci portálu.

3.8.3 Administrátorská část

Administrátorská část je umístěna na specifické adrese tak, aby o ní běžní uživatelé nevěděli. Do této sekce mají přístup jen vybraní administrátoři, jimž slouží pro kompletní správu obsahu na webovém portálu. Administrace (Příloha D 7) se skládá z těchto stránek:

- **Souhrn** – stránka, kterou administrátoři uvidí ihned po přihlášení a na které vidí rychlý souhrn o nejnovějších registracích uživatelů portálu, zahájených reklamacích, vložených firmách a hodnoceních.
- **Editor** – pomocí této stránky je možné spravovat většinu z textů stránek zobrazených na webu. Po zvolení konkrétní položky v menu je administrátorovi ve

WYSIWYG editoru zobrazen obsah stránky tak, jak jej vidí koncový uživatel. Pomocí tohoto editoru administrátor může libovolně pracovat s obsahem.

- **Uživatelské účty** – v této sekci probíhá kompletní správa uživatelských účtů. Administrátor zde vidí detailní informace o uživateli, například počet vložených dokladů a reklamací či údaje o posledním přihlášení. Zároveň má také možnost jednotlivé účty vytvářet, editovat, mazat a případně blokovat.
- **Doklady** – sekce pro kontrolu vložených dokladů.
- **Reklamace** – sekce pro kontrolu vložených reklamací.
- **Firmy** – zde jsou zobrazeny veškeré firmy, které mají v rámci portálu vytvořený profil. Uživatelská role Obchodník zde má možnost upravovat pozice v žebříčku firem. Administrátoři se zde starají o to, aby všechna firemní data (loga a popisy) byla aktuální.
- **Hodnocení firem** – v rámci této sekce mají administrátoři možnost zobrazovat, schvalovat či zamítnout uživatelské hodnocení firem.
- **Poradna** – hlavní sekce pro správu poradny. Administrátoři zde mají možnost vytvářet nové články (otázky) v poradně.
- **Poradna kategorie** – v této sekci administrátoři spravují jednotlivé kategorie poradny. V rámci každé kategorie je možné upravování klíčových slov.
- **Statistiky** – pomocí této stránky je možné sledovat klíčové statistiky. Administrátor si pouze zvolí rozsah dat, v jakých se mají statistiky zobrazit (výchozí hodnota je 14 dní), a systém automaticky statistiky přepočítá.
- **Šablony emailů** – stránka pro úpravu odesílaných emailových šablon.
- **Administrátorské účty** – Sekce pro správu účtů administrátorů. U každého účtu je možné nastavit jeho roli v administraci (role definované v kapitole 2.1.5).

3.8.4 Automatizovaný skript

Poslední částí webového portálu je automatizovaný skript, který se pomocí Cronu spouští pravidelně v nočních hodinách (v době, kdy je na internetu nejmenší provoz a webový portál je nejméně vytížen). Tento skript pokaždé nalezne veškeré produkty, které dosáhly požadovaného termínu pro notifikaci (expirace období reklamace produktu a blížící se termín pro vyřízení reklamace). U nalezených produktů následně zašle uživatelům email informující o těchto termínech.

4 Doplnění dalších funkcí prostřednictvím aplikací pro mobilní zařízení

Chytrá mobilní zařízení se v posledních letech na trhu velice rozšířila. Široké spektrum uživatelů internetu používá minimálně chytrý telefon či tablet. Proto by vývojáři webových portálů neměli opomenout význam těchto zařízení. [5]

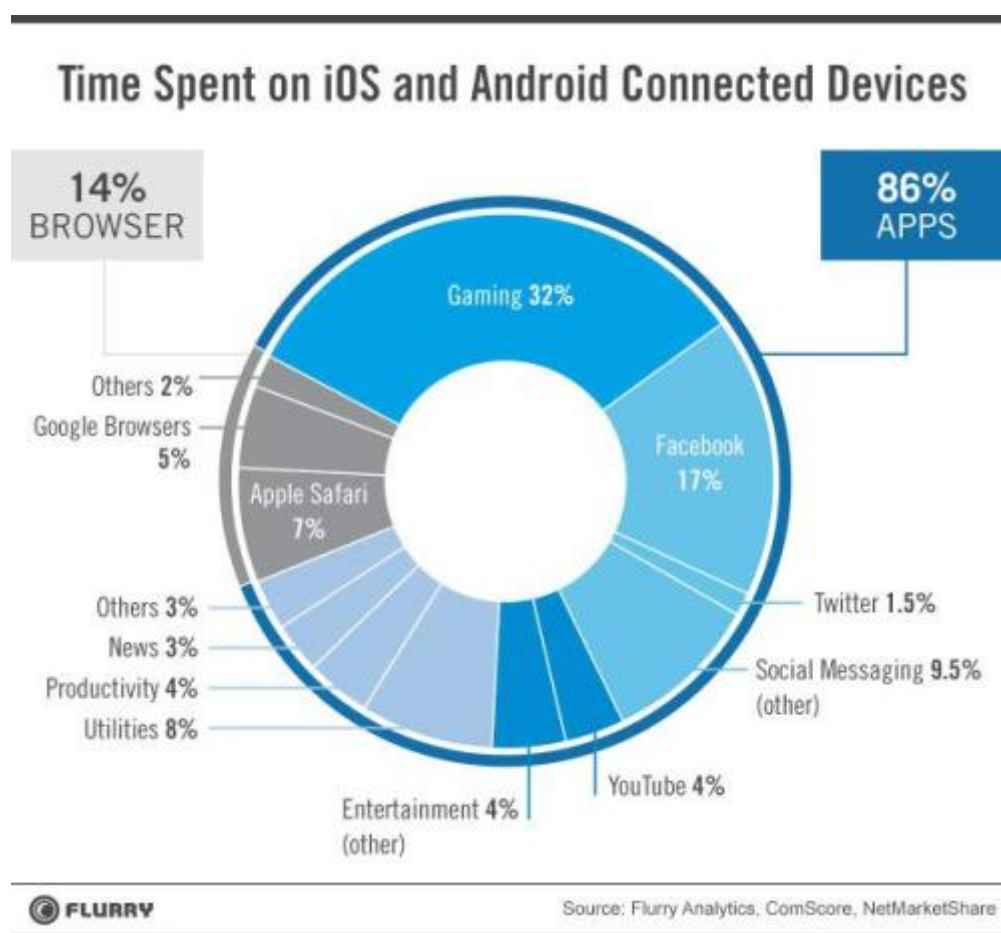
Na trhu je dnes rovněž široké spektrum rozličných mobilních operačních systémů. Pokud by měl vývojář vyvíjet aplikaci pro každou z těchto platform, pravděpodobně by strávil velké množství času optimalizací aplikace pro systémy, které využívá jen hrstka uživatelů. Toto řešení je ze své podstaty velmi neefektivní, proto se vývojáři orientují na dominantní operační systémy, z nichž nejpoužívanějšími jsou:

- **Android** – systém vyvíjený společností Google, postavený na linuxovém jádře a distribuovaný jako open source. Mezi výrobci mobilních zařízení je oblíbený díky své cenové politice, nízkým hardwarovým nárokům a též díky možnosti tvorby vlastních nadstaveb (například TouchViz vyvíjený společností Samsung). Vývoj aplikací probíhá v jazyce Java.
- **iOS** – systém od společnosti Apple, který je dostupný pouze pro zařízení vyráběná touto firmou. Původně byl vytvořen pro revoluční telefon iPhone a následně se dostal i do dalších vznikajících zařízení (iPad, iPod apod.). Pro vývoj je využíván programovací jazyk Apple Objective-C.
- **Windows Phone** – systém firmy Microsoft nahrazující systém Windows Mobile a Zune. Jeho nasazení je možné i na zařízeních od jiných výrobců⁴, nicméně jeho uživatelská obliba roste pomalu. Oproti Androidu a iOS je mu často vytýkáno malé množství dostupných aplikací. V rámci podporovaných programovacích jazyků je možné použít C# a Visual basic.

Webové portály vznikající v rámci dnešního trhu jsou ve většině případů optimalizovány pro mobilní zařízení, mnoho z nich začíná k podpoře podnikání využívat mobilních aplikací. Dle grafu (Obrázek 9) „Rozložení časů při používání

⁴ Microsoft je díky akvizici společnosti Nokia, jež byla původně hlavním producentem zařízení využívajících Windows Phone, považován za výrobce těchto zařízení.

mobilních zařízení užívateli iOS a Android“ z roku 2014 je vidět, že uživatelé tráví až 86 % stráveného času na zařízení používáním mobilních aplikací, a pouze 14 % používáním mobilních webových prohlížečů. Na základě této statistiky je vhodné doplnit funkce webového portálu tak, aby uživatelé nemuseli používat jen mobilní prohlížeč, ale i vyvinutou mobilní aplikaci.



Obrázek 9: Graf rozložení času při používání mobilních zařízení užívateli iOS a Android

Zdroj: <http://flurrymobile.tumblr.com/post/115191864580/apps-solidify-leadership-six-years-into-the-mobile>

4.1 Analýza předpokladů

Vývojáři dnes mají široké možnosti podpory webových portálů pomocí mobilních aplikací. Někteří z nich vyvíjejí aplikace se stejnou funkčností, jakou má webový portál, přičemž jejich funkcionalita je provázána pomocí API. Další formou aplikací jsou například různé hry a interaktivní prezentace, které pomáhají posilovat povědomí o značce a získávat další

potenciální uživatele. Současně mohou aplikace sloužit jako přímý marketingový kanál pro distribuci informací (například dostupné termíny, ceny a doplňující informace).

Při porovnání s webovými portály vyplývají z používání mobilních aplikací tyto hlavní přínosy:

- **Rozsáhlá uživatelská komunita** – hlavní mobilní platformy dnes mají rozsáhlé množství aktivních uživatelů.
- **Používání off-line** – uživatel nemusí být pro používání aplikace připojen k internetu. Pokud jsou v rámci aplikace synchronizována data s webovým portálem, je možné se synchronizaci počkat do doby, kdy se zařízení k internetu znovu připojí.
- **Využití hardwarových funkcí zařízení** – oproti webu je možné snadno využívat např. GPS, akcelerometr či mikrofon.
- **Zpoplatnění** – vývojář mobilních aplikací má (oproti vývojáři webových portálů) možnost si nechat aplikaci od uživatelů zaplatit. Zpravidla se používá obchodní model, kdy je základní aplikace zdarma, ale na rozdíl od placené verze obsahuje reklamy nebo nemá kompletní funkčnost.

I přes uvedené přínosy mají mobilní aplikace oproti webovým portálům některé nevýhody:

- **Dražší náklady na vývoj a úpravy** – je potřeba vyvinout aplikaci pro všechny hlavní mobilní operační systémy a nad nimi provést následné otestování.
- **Správa verzí** – je nutností, aby si všichni uživatelé pravidelně stahovali aktuální verzi aplikace. Tento proces je většinou automatický, nicméně dojde-li k rozšíření funkcností, díky které začne aplikace využívat další funkce telefonu (např. ovládání fotoaparátu, či přístup ke kontaktům), je nutné provést manuální schválení.
- **Pomalé zveřejňování** – při nahrávání nové verze aplikace do obchodů s aplikacemi je nutné projít celým schvalovacím procesem, který se zabývá i revizemi zdrojového kódu.

V rámci projektu Reklamář.cz bylo rozhodnuto o vzniku mobilní aplikace, která umožní uživatelům používat téměř všechny funkce, jaké mají na webu (práce s doklady, reklamacemi, přístup k hodnocení firem a poradny). Modul poradny a hodnocení firem bude součástí aplikace, přičemž jeho obsah bude dostupný pouze při připojení k internetu –

to z toho důvodu, že při stahování celého obsahu těchto sekcí by došlo k velkému vytížení datové uložičtě mobilního zařízení.

4.2 Výběr vhodné technologie

Na trhu se vyskytuje široké spektrum technologií pro vývoj mobilní aplikací, přičemž každá z nich je vhodná pro odlišná zařízení a požadovanou funkčnost. Webový vývojáři se proto musí rozhodnout, zda budou pro každou platformu vyvíjet aplikace v rámci individuálního prostředí nebo zda použijí řešení multiplatformní. Výhodou druhých jmenovaných řešení je fakt, že není nutností vytvářet několik identických aplikací – stačí vyvinout pouze jádro, nad nímž vzniknou individuální komunikační rozhraní pro každý mobilní operační systém. Tato práce se bude dále zabývat právě multiplatformním vývojem. [6]

Mezi nepoužívanější multiplatformní vývojová prostředí patří:

- **Xamarin** – nástupce projektu Mono.⁵ Vývoj aplikací probíhá v programovacím jazyce C# a platformě .NET. Pomocí Xamarinu lze vyvíjet i složitější aplikace pro všechny tři hlavní operační systémy.⁶
- **Apache Cordova** – umožňuje tvorbu multiplatformních aplikací pro většinu dostupných mobilních operačních systémů. Aplikace jsou vyvíjeny pomocí HTML, CSS a javascriptu.
- **PhoneGap** – technologie od firmy Adobe s open source licencí. Stejně jako u Cordovy jsou aplikace vyvíjeny pomocí HTML, CSS a JavaScriptu.

Na základě provedených analýz se autor rozhodl zvolit prostředí Xamarin, a to ze dvou důvodů: zaprvé kvůli zkušenostem autora s platformou .NET a zadruhé kvůli potřebě vytvořit výkonnou nativní aplikaci.

Xamarin byl pro vývojáře původně zpoplatněný, nicméně studentům licenční politika umožňovala (po ověření statutu studenta) volně využívat jeho nástroje. V průběhu psaní této práce došlo k akvizici Xamarinu Microsoftem a změně licenční politiky. Nyní je umožněno využití tohoto nástroje zdarma pro všechny uživatele vývojového prostředí

⁵ Mono umožňovalo tvorbu aplikací pro iOS a Android pomocí platformy .NET.

⁶ Viz strana 79.

Microsoft Visual Studio, včetně komunitní edice (tato edice je zdarma pro individuální vývojáře, malé týmy a pro školní prostředí). [32]

4.3 Návrh aplikace dle zvolené technologie

Vzhledem k faktu, že v rámci této fáze již existoval webový portál a byla jasně dána jeho kritéria, bylo možné definovat požadavky pro vznikající mobilní aplikaci.

Mobilní aplikace portálu Reklamář.cz by měla umožňovat:

- Snadné a intuitivní ovládání
- Napojení na připravené API
- Zasílání upozornění do notifikační lišty
- Použití fotoaparátu integrovaného v zařízení pro focení dokladů

Při spuštění bude aplikace vyžadovat přihlášení, případně registraci. Poté bude uživateli zobrazena úvodní stránka s logem a s ikonou pro nastavení aplikace, dále rozcestník čtyř hlavních sekcí (Doklady, Reklamace, Hodnocení firem a Poradna). V rámci každé sekce bude zachována aplikační logika, kterou uživatelé znají z webového portálu.

4.4 Vývoj

Při zahájení vývoje byl vytvořen projekt pro Android, do něhož bylo začleněno rozhraní pro Windows Phone. Pro možnost vývoje aplikace zároveň i pro iOS, bylo nutné zajistit připojení počítače s operačním systémem Mac OS, a nainstalovaným prostředím Xamarin do lokální sítě. Poté bylo možné začít reálně vyvíjet aplikaci.

Nejprve dle specifikovaného návrhu vzniklo rozhraní, v jehož rámci byly vytvořeny jednotlivé ovládací prvky, byla doplněna grafika a vytvořeny formuláře. K tomuto rozhraní byly postupně přidány formuláře a nástroje na zpracování dat. Zároveň došlo k tvorbě komunikačního rozhraní (které bude sloužit pro komunikaci s API webového portálu).

4.5 Testování

Aplikace je v době odevzdání diplomové práce na pomezí fáze vývoje a testování. S ohledem na plné časové vytížení autorů portálu Reklamář.cz a na postupné učení se v rámci vývoje mobilních aplikací je spuštění portálu i mobilní aplikace odloženo.

Mobilní aplikace lze testovat třemi možnými způsoby:

- **Testy na reálných zařízeních** – nejlepší možný způsob testování, při kterém je možné otestovat kompletní funkcionalitu. Vývojář však musí mít k dispozici dostatečný počet reálných zařízení.
- **Testy na virtuálních zařízeních** – tento způsob řeší potřebu mnoha testovacích zařízení jejich virtualizací. V rámci PC, na kterém je aplikace vyvíjena, může vývojář otestovat širokou škálu zařízení. Úskalí virtuálních testů spočívá v tom, že není možné využít všechny možnosti reálných zařízení.
- **Kombinované testy** – vzhledem k úskalím předchozích typů testů se ideálním případem jeví kombinace testů na reálných i virtuálních zařízeních zároveň. Je dobré aplikaci otestovat nejprve v rámci virtuálního prostředí a po akceptaci přejít k testům na reálných zařízeních.

V rámci testů je nutné kromě testování funkčnosti a výkonosti i testování uživatelské přívětivosti. Vývojáři musí myslet na to, že aplikace bude používána na různých zařízeních (od malých telefonů až po velké tablety) a je proto nutné tomu přizpůsobit ovládání. Pokud se uživatel nedokáže v aplikaci zorientovat, či je pro něj obtížné její používání, je nutné aplikační logiku upravit. [6]

4.6 Publikování aplikací

Pokud je aplikace otestována a odladěna, je připravena pro distribuci k uživatelům. K té slouží unikátní oficiální obchody v rámci každého mobilního operačního systému. Každý obchod má odlišná specifika, s čímž souvisí fakt, že nahrávané aplikace musí splnit odlišné předpoklady.

Po publikování se aplikace dostává mezi uživatele. Nyní záleží na tom, jaké možnosti propagace nových aplikací jejich vlastníci zvolí: například na webových portálech je možné využít možnosti umístit odkazy směřující uživatele do těchto obchodů (formou mohou být kromě klasických odkazů i QR kódy).

Publikování aplikací pro jednotlivé operační systémy probíhá v následujících obchodech:

- **Android: Play Store** – Pro publikování je nutné mít zaregistrován publikační účet se zaplaceným vstupním poplatkem. Uživateli s ověřeným účtem (vývojáři) je

umožněno za pomoci intuitivního nástroje aplikaci nahrát a doplnit příslušné informace. Poté je aplikace dostupná všem uživatelům.

- **iOS: App Store** – Zde je proces nahrávání složitější, neboť je nutné splnit předem požadovaná kritéria. Publikování probíhá pomocí nástroje iTunes Connect. Nejprve je nutné vytvořit distribuční certifikát a opravňující účet. Poté je možné aplikaci nahrát a při splnění certifikačního procesu uvolnit k používání uživatelům.
- **Windows Phone: Store** – Zde je situace nejjednodušší. Pro publikování stačí mít pouze aktivní Microsoft účet, přičemž po nahrání aplikace a vytvoření profilu v obchodu je aplikace veřejně dostupná.

4.7 Zhodnocení vývoje mobilní aplikace

Postupně realizovaná mobilní aplikace by mohla být pro portál Reklamář.cz velice přínosná. Díky plánovanému snadnému, intuitivnímu ovládání budou uživatelé moci rychle a jednoduše přidávat nové doklady přímo při nákupu v obchodech. Pomocí využití notifikačního centra v mobilních zařízeních budou uživatelé vždy včas informováni o všech potřebných informacích týkajících se jejich dokladů a reklamací.

5 Zhodnocení výsledků projektu

Tato kapitola má za úkol shrnout časovou i finanční náročnost projektu Reklamář.cz. Současně budou zváženy náklady na alternativní způsob vývoje.

5.1 Technické přínosy

Realizovaný webový portál odpovídá požadavkům definovaným v kapitole 2. Skládá se ze tří hlavních modulů, které (prozatím několika vybraným) uživatelům pomáhají se správou dokladů a reklamací, s hodnocením firem a s poradenstvím. Přihlašování pomocí Facebooku zjednodušuje přístup do klientské sekce (pokud je uživatel v rámci webového prohlížeče již na Facebook přihlášen, nemusí v rámci portálu vyplňovat přihlašovací údaje, pouze klikne na tlačítko). Proces obnovy hesla je funkční a bezpečný. Díky automatickým notifikacím již byl několikrát zrealizován reklamační proces, v jehož rámci došlo k úspěšnému vyřešení reklamace. Dalším úspěchem byla pozitivní zpětná vazba od uživatelů používající při zadávání firem rozhraní pro komunikaci s portálem ARES. Uživatelé si chválili především snadnost používání tohoto nástroje.

Použitý redakční systém funguje spolehlivě. Díky poskytnutým nástrojům umožňuje snadnou správu obsahu webového portálu včetně správy uživatelských dat.

Podpora mobilních zařízení pomocí responsivního zobrazení je pro uživatele též vítaným přínosem. Díky širokému spektru různých zařízení (tablety, mobilní telefony) má projekt pozitivní zpětnou vazbu z pohodlného používání.

5.2 Ekonomická a časová analýza

Vývoj a nasazení samotného webového portálu zabraly několikačlennému vývojovému týmu 184 hodin. Jednotlivé etapy a jejich časová náročnost jsou zobrazeny na obrázku 10, ze kterého je patrné, že největší podíl zabral část vývoje (Návrh zvoleného řešení) – celkem 110 hodin. Při zavádění systému se předpokládá nárůst časové náročnosti u položky zkušebního období.

Alternativou vlastního vývoje by bylo zadání tvorby portálu specializované firmě, která by zajistila vývoj, nasazení a otestování. Při průměrném nákladu webového vývojáře 400 Kč

na hodinu by výsledná cena za zrealizovaný portál, s předpokládaným rozsahem prací 168 hodin⁷, činila 67 200,- Kč. [33]

Časová analýza vývoje webového portálu Reklamář.cz



Obrázek 10: Časová analýza vývoje portálu Reklamář.cz

Zdroj: vlastní

Výsledné náklady realizace webového portálu byly vynaloženy pouze na webhosting (363,- Kč / rok), CZ doménu (151,- Kč / rok) a SSL certifikát (490,- Kč / rok). Protože v rámci projektu nebyla kupována žádná placená komponenta a použité nástroje pro vývoj byly neplacené, další náklady technické stránky byly nulové.

Náklady na vývoj mobilní aplikace jsou zatím též nulové, neboť použité vývojové prostředí je zdarma a vývoj probíhá vlastními silami. Současně není potřeba využívat žádných placených modulů.

5.3 Budoucnost projektu

Po dokončení mobilní aplikace včetně jejího napojení na připravované API bude možné portál veřejně spustit (plánovaný termín je přelom července a srpna 2016). Vzhledem k nenalezení žádné zásadní chyby i přes provedení mnoha nezávislých testů není očekáván vznik dalších vícenákladů na vývoj portálu. Pokud by se chyba vyskytla, bude nutná její důkladná analýza a rychlá oprava. Po spuštění může dojít k potřebě rozšířit funkci portálu – například na základě požadavků uživatelů či s ohledem na vznikající konkurenci. S touto eventualitou autoři projektu počítají, vzniklá dokumentace a strukturovaný zdrojový kód jim případně umožní portál snadno rozšířit.

⁷ Fáze plánování jsou vynechány, neboť jsou vnímány jako zadání projektu.

Závěr

Diplomová práce si kladla za úkol navrhnout a dle návrhu (s přihlédnutím k aktuálním trendům v oblasti vývoje aplikací) následně vyvinout moderní webový portál. Zrealizovaný portál měl být (a je) dle zadání optimalizovaný nejen pro PC, ale i pro mobilní zařízení.

V úvodu práce autor nejprve rozčlenil nejrozšířenější technologie pro vznik webových stránek do tří vrstev webového obsahu, následně na tomto základě definoval obecný pojem moderní webový portál. Aby webovému portálu příslušelo přívlastko moderní, měl by využívat vlastností technologií HTML5, CSS3 a Javacriptu, jejichž zkombinování umožňuje nejen vytvářet pokročilé webové portály, ale také nahradit stárnoucí a nezabezpečený Flash. Za jednu ze základních vlastností moderního portálu lze považovat podporu mobilních zařízení, pro kterou je nezbytný tzv. responsivní přístup k tvorbě struktury webových stránek. Tento přístup umožňuje dynamicky upravovat uskupení a viditelnost jednotlivých prvků webových stránek pro plné využití dostupného rozlišení libovolného zařízení. V rámci trendu zvyšování zabezpečení webových portálů bylo v práci zohledněno i zabezpečení správy hesel a databází či využití HTTPS, které znemožňuje odposlech komunikace mezi uživatelem a webovým serverem. Pro uskutečnění praktické části této práce bylo třeba definovat pokročilejší přístupy k tvorbě databází a zvolit vhodnou metodiku vývoje (v tomto případě metodiku SDLC).

Dalším cílem práce byla analýza vybraného projektu. Práce se zaměřuje na analýzu webového portálu Reklamář.cz spojenou s jeho vývojem. Tento vznikající portál je v rámci prostředí českého internetu unikátní, neboť uživatelům nabízí služby v oblastech procesu reklamace, správy účtenek zakoupených produktů a expirací lhůt, v nichž jsou produkty reklamovatelné. Dále umožňuje prohlížení ucelené znalostní báze a poskytuje uživatelům možnost hodnocení evidovaných firem v rámci realizovaných reklamací, tedy i celkovou zpětnou vazbu pro všechny potenciální zákazníky těchto firem. Analýzy byly zaměřeny na požadovanou funkčnost portálu a na plánované technické prostředky; na jejich základě bylo možné definovat očekávanou funkčnost a podrobná specifika (např. definice uživatelských rolí, či plánované napojení na ARES).

Na základě provedených analýz a předem určených požadavků byl zrealizován vývoj webového portálu. Při vývoji došlo ke splnění požadavků definovaných v teoretické části této práce. V rámci podpory mobilních zařízení je web responsivní a splňuje nároky pro ovládání na dotykových zařízeních. Současně poskytuje API pro komunikaci se vznikající mobilní aplikací. Z důvodu plného časové vytížení autorského týmu není zatím portál volně dostupný všem uživatelům internetu, ale pouze uživatelům vybraným. Plánované spuštění pro veřejnost je naplánováno na přelom července a srpna roku 2016.

V předposlední kapitole byly zanalyzovány možnosti vývoje mobilní aplikace, která by doplnila další funkce webového portálu. Hlavní přednosti mobilních aplikací spočívají pro uživatele v možnosti jejich využívání i v momentech, kdy není zařízení připojeno k internetu, a také ve snadném přístupu k aplikaci bez nutnosti použití mobilního prohlížeče webových stránek. Právě tyto výhody vedly autora práce k rozhodnutí o vývoji mobilní aplikace. Pro vývoj bylo zvoleno vývojové prostředí Xamarin, které umožňuje vyvinout jedinou aplikaci pro tři hlavní operační systémy mobilních zařízení najednou. Mobilní aplikace je v době odevzdání této práce ve fázi pokročilého vývoje a testování. Pro uživatele webového portálu Reklamář.cz bude dostupná v budoucnu.

Autor v práci vychází z nejaktuálnějších trendů ve vývoji webových portálů. Specifickou vlastností oboru IT je neustálý, dynamický vývoj různých technologií i přístupů. Postupně bude docházet ke schvalování nových specifikací HTML5 a CSS3 a díky tomu bude možné vytvářet pokročilejší aplikace, které postupně z trhu vytlačí technologii Flash. Díky neustálému vývoji programovacích jazyků i vzniku rozšiřujících knihoven pro JavaScript lze očekávat převod klasických desktopových aplikací do online prostředí. Neustálý vývoj výpočetních kapacit a technik pro útoky na webové stránky s sebou do budoucna ponese nutnost neustálého zajišťování co nejvyšší míry zabezpečení pro webové portály.

Seznam použité literatury

Citace

[1] GASSTON, Peter. 2015. *Moderní web*. Překlad Ondřej Baše. Brno: Computer Press. ISBN 9788025143452.

[2] PILGRIM, Mark. 2015. *Ponořme se do HTML5*. Praha: CZ.NIC, z.s.p.o. CZ.NIC. ISBN 9788090580268.

[3] ŘEZÁČ, Jan. 2014. *Web ostrý jako břitva: návrh fungujícího webu pro webdesignery a zadavatele projektů*. Jihlava: Baroque Partners. ISBN 978-80-87923-01-6.

[4] Doyle, R. *Modern web application systems: What it really means*. 2013. *The Secured Lender* [online]. [cit. 2016-03-30]. Dostupné z:
<http://search.proquest.com/docview/1723932499?accountid=17116>

[5] LACKO, Luboslav. 2015. *Vývoj aplikací pro Android*. Brno: Computer Press, 472 s. ISBN 9788025143476.

[6] PETZOLD, Charles. *Creating Mobile Apps with Xamarin.Forms*. Preview edition 2. Redmond, WA: Microsoft Press, 2015, 453 p. ISBN 978-0-7356-9723-2.

[7] How does the Internet work. 2014. *W3C* [online]. [cit. 2016-03-30]. Dostupné z:
https://www.w3.org/wiki/How_does_the_Internet_work

[8] DB-Engines Ranking. 2016. *Solid IT* [online]. [cit. 2016-04-01]. Dostupné z: <http://db-engines.com/en/ranking>

[9] Usage of server-side programming languages for websites. 2016. *W3Techs* [online]. [cit. 2016-04-01]. Dostupné z:
http://w3techs.com/technologies/overview/programming_language/all

[10] Modern. (n.d.). 2016. *Merriam-Webster* [online]. [cit. 2016-04-01]. Dostupné z:
<http://www.merriam-webster.com/dictionary/modern>

- [11] MCINTIRE, Penny. *Visual Design for the modern web*. Berkeley, CA: New Riders, c2008. ISBN 0321515382.
- [12] The 8 Elements of Modern Web Design (And Web Design Trends to Watch). 2015. *Hubspot: blog* [online]. [cit. 2016-04-01]. Dostupné z: <http://blog.hubspot.com/marketing/elements-of-modern-web-design-list>
- [13] GASSTON, Peter. *The book of CSS3: a developer's guide to the future of web design*. Second edition. San Francisco: No Starch Press, 2015. ISBN 978-1-59327-643-0.
- [14] DUCKETT, Jon. c2011. *HTML & CSS: design and build websites*. Indianapolis, IN: Wiley. ISBN 1118008189.
- [15] The Modern JavaScript Developer's Toolbox. 2015. *InfoQ* [online]. [cit. 2016-04-01]. Dostupné z: <http://www.infoq.com/articles/modern-javascript-toolbox>
- [16] Usage of JavaScript libraries for websites. 2016. *W3Techs* [online]. [cit. 2016-04-01]. Dostupné z: http://w3techs.com/technologies/overview/javascript_library/all
- [17] CHAFFER, Jonathan a Karl SWEDBERG. *Mistrovství v jQuery: [kompletní průvodce vývojáře]*. Brno: Computer Press, 2013. Mistrovství. ISBN 978-80-251-4103-8.
- [18] KADLEC, Tim. *Responzivní design profesionálně*. Brno: Zoner Press, 2014. Encyklopedie Zoner Press. ISBN 978-80-7413-280-3.
- [19] CLARK, Jason A. *Responsive web design in practice*. Lanham: Rowman & Littlefield, 2015. ISBN 9781442243699.
- [20] DE RYCK, Phillippe. 2014 *Primer on client-side web security*. New York: Springer, 2014. ISBN 9783319122250.
- [21] The 25 Most Popular Passwords of 2015. 2016. *Gizmodo* [online]. [cit. 2016-04-01]. Dostupné z: <http://gizmodo.com/the-25-most-popular-passwords-of-2015-were-all-such-id-1753591514>
- [22] BRAMER, Max. 2015. *Web programming with PHP and MySQL: a practical guide*. New York, NY: Springer Berlin Heidelberg. ISBN 9783319226583.

- [23] CHERRY, Denny. *Securing SQL server: protecting your database from attackers*. Third edition. Waltham, MA: Elsevier/Syngress, 2015. ISBN 9780128012758.
- [24] AULD, Chris. *Practical Database Design for the Web*. Berkeley, CA: Apress, 2002. ISBN 9781430253778.
- [25] BRUCKNER, Tomáš, Jiří VOŘÍŠEK, Alena BUCHALCEVOVÁ, Iva STANOVSKÁ, Dušan CHLAPEK a Václav ŘEPA. *Tvorba informačních systémů: principy, metodiky, architektury*. První vydání. Praha: Grada Publishing, 2012. Management v informační společnosti. ISBN 978-80-247-4153-6.
- [26] BOYDE, Joshua. *A down-to-earth guide to SDLC project management: getting your system development life cycle project successfully accross the line using PMBOK-- in an adaptive way*. [S.l.: CreateSpace Independent Pub. Platform], 2012. ISBN 9781480038196.
- [27] VRANA, Ivan a Karel RICHTA. *Zásady a postupy zavádění podnikových informačních systémů: praktická příručka pro podnikové manažery*. Praha: Grada, 2005. Management v informační společnosti. ISBN 80-247-1103-6.
- [28] ROUDENSKÝ, Petr a Anna HAVLÍČKOVÁ. *Řízení kvality softwaru: průvodce testováním*. Brno: Computer Press, 2013. ISBN 978-80-251-3816-8.
- [29] Šanon. 2016. *Airbank* [online]. [cit. 2016-04-01]. Dostupné z: <https://www.airbank.cz/cs/bezny-ucet/sanon/sanon/>
- [30] Jak jsme zlepšili zabezpečení Slevomatu. 2014. *Michal Špaček* [online]. [cit. 2016-04-16]. Dostupné z: <https://www.michalspacek.cz/prednasky/jak-jsme-zlepsili-zabezpeceni-slevomatu-pixdevday>
- [31] MATTHEWS, Andy a Sahne GLISER. 2015. *Creating Mobile Apps with jQuery Mobile*. Second edition. Livery Place: Pack Publishing Ltd. ISBN 978-1-78355-511-6.
- [32] Microsoft Makes Xamarin Free For Visual Studio Customers. *MSPoweruser* [online]. 2016 [cit. 2016-04-16]. Dostupné z: <http://mspoweruser.com/microsoft-makes-xamarin-free-for-visual-studio-customers/>

[33] *Průzkum platů 2013/14: Přehled plateb dle regionů v ČR* [online]. 2014. In: . Grafton recruitment, s. 58 [cit. 2016-04-15]. Dostupné z:

www.grafton.cz/~media/Files/Downloads/CZ/Salary%20Survey%202013_14.ashx

Elektronická databáze článků ProQuest (knihovna.tul.cz)

Seznam příloh

Příloha A – Úvodní stránka portálu

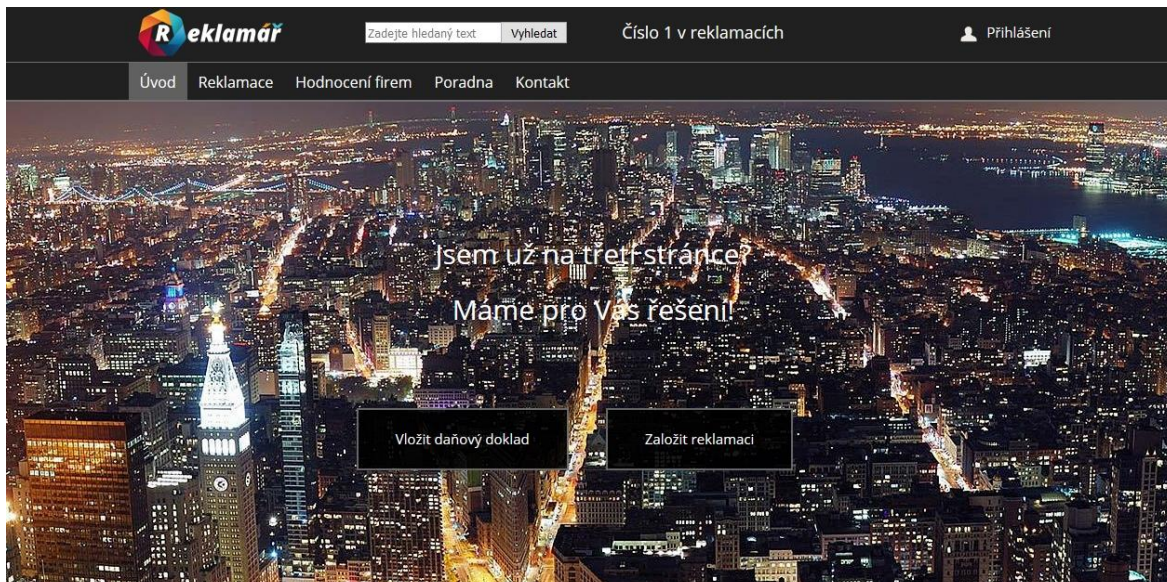
Příloha B – Zobrazení úvodní stránky při změně horizontálního rozlišení

Příloha C – Záznamy z testování

Příloha D – Vybrané stránky portálu

Příloha A – Úvodní stránka portálu

1. Hrubý koncept návrhu portálu



Reklamář řeší reklamační proces za Vás - zcela ZDARMA!

1. KROK
Vložení dokladu po nákupu produktu u obchodníka



2. KROK
Zadání reklamačního procesu při problému během záruční lhůty



3. KROK
Automatické hledání vyřízení reklamace a hodnocení obchodníka



Každý se někdy setkal s problémem, kdy zakoupený produkt přestal během záruční doby fungovat. Jakmile se tato nepříjemnost stane, začíná dlouhý proces vyřizování reklamace, který s sebou často přináší velké časové zatížení.

Kde mám uložené doklady o koupi? Jak vlastně řešit reklamaci a na co mám během procesu právo?

S naším systémem již nemusíte nic z toho řešit. Jednoduše Vás provedeme procesem reklamace a když si nebudete vědět rady, v naší poradně dostanete na všechny otázky jasné odpovědi.

Hlavní funkce systému Reklamář

- **Evidence dokladů, faktur v elektronické formě** - nemusíte složitě hledat doklady, když jsou potřeba, automatické upozornění na konec záruční doby produktu
- **Proces reklamace** - založení reklamace, hlídání stavu vyřízení, automatické upozornění prodejce o Vaší reklamaci, všechny potřebné dokumenty k reklamaci, rady v případě nesnází
- **Hodnocení firem** - jsme první portál, který hodnotí obchody dle jejich chování k zákazníkům v průběhu reklamačního procesu, ne pouze při nákupu, kdy vše funguje
- **Rozsáhlý manuál pro spotřebitele** - na jednom místě naleznete všechny potřebné informace k reklamačnímu procesu, práva a povinnosti spotřebitele, novinky a zajímavé typy



Nejlepší zkušenosti



Reklamář

Web

Info

O nás

2. Finální podoba portálu na testovacím serveru

Pomůžu Vám s každou reklamací...

Vyhledat

Přihlášení

Úvod
Reklamacie
Hodnocení firem
Poradna
Kontakt

Ověřte si svá práva a povinnosti

Spotřebitelská poradna shrnuje všechny informace o reklamacích

Vložit daňový doklad

Založit reklamaci

Vytvořte reklamaci ve třech krocích

- 1. Vložení dokladu o nákupu zboží od obchodníka**

Doklad v elektronické formě již nemusíte fyzicky archivovat. Reklamář Vás bude informovat o končící době záruky zboží, takže již nikdy neprosvíhnete šanci na reklamaci.
- 2. Založení nové reklamace v případě vzniku problému**

Stačí vybrat uložený doklad s problematickým zbožím, přidat způsob předání obchodníkovi a popsat vzniklou vadu. Další kroky již budou automaticky vyřešeny za Vás.
- 3. Automatické vytvoření potřebných dokumentů a hlídání vyřízení**

Reklamář připraví průvodní dopis, štítky pro poštu, informuje prodejce a ohlídá zákonnou dobu na vyřízení reklamace. Při problémech s výsledkem Vám poradí naše poradna.

„Garantujeme pohodlné sledování vyřízení Vaší reklamace.“

Každý se již někdy setkal s problémem, kdy zakoupený produkt přestal během záruční doby fungovat. Jakmile dojde k této nepřijemnosti, začíná dlouhý proces vyřizování reklamace, který s sebou často přináší velké časové zatížení.

S naším systémem Reklamář snížíte časovou zátěž na minimum. Jednoduše Vás provedeme procesem reklamace a když si nebudete vědět rady, v naší poradně dostanete na všechny otázky jasné odpovědi.

Nejlepší hodnocení

OKEY s.r.o.

★ ★ ★ ★ ☆

OKEY s.r.o.

★ ★ ★ ★ ☆

OKEY s.r.o.

★ ★ ★ ☆ ☆

OKEY s.r.o.

★ ★ ☆ ☆ ☆

OKEY s.r.o.

★ ☆ ☆ ☆ ☆

OKEY s.r.o.

★ ☆ ☆ ☆ ☆

Nejčastější dotazy

Jaká má spotřebitel práva, objeví-li se vada v záruční lhůtě?

Prodejce odmítá přijmout reklamaci a odkazuje mne na výrobce

Jaká má spotřebitel práva, objeví-li se vada v záruční lhůtě?

Nejnovější hodnocení uživatelů

OKEY s.r.o.

Luboš Tomešek, 14.4. 12:10

★ ★ ★ ★ ☆

Hodnocení: Neuznaná reklamace i když nebyl důvod. Arogantní jednání a ...

Jablotron Alarms a.s

Ondřej Janoušek, 14.4. 10:15

★ ★ ★ ★ ☆

Hodnocení: Reklamace bez problému. Produkt vrácen úspěšně...

OKEY s.r.o.

Luboš Tomešek, 14.4. 6:50

★ ★ ★ ★ ☆

Hodnocení: Reklamace byla vyřízena rychle (15 dní). Prodejce jednal solidně...

Reklamář

TOMEPOL s.r.o.
Přepeře 162
512 61 Přepeře
Česká republika

Web

Úvod
Reklamacie
Hodnocení firem
Poradna
Kontakt

Info

Facebook

Copyright (©) 2016
CabaDesign

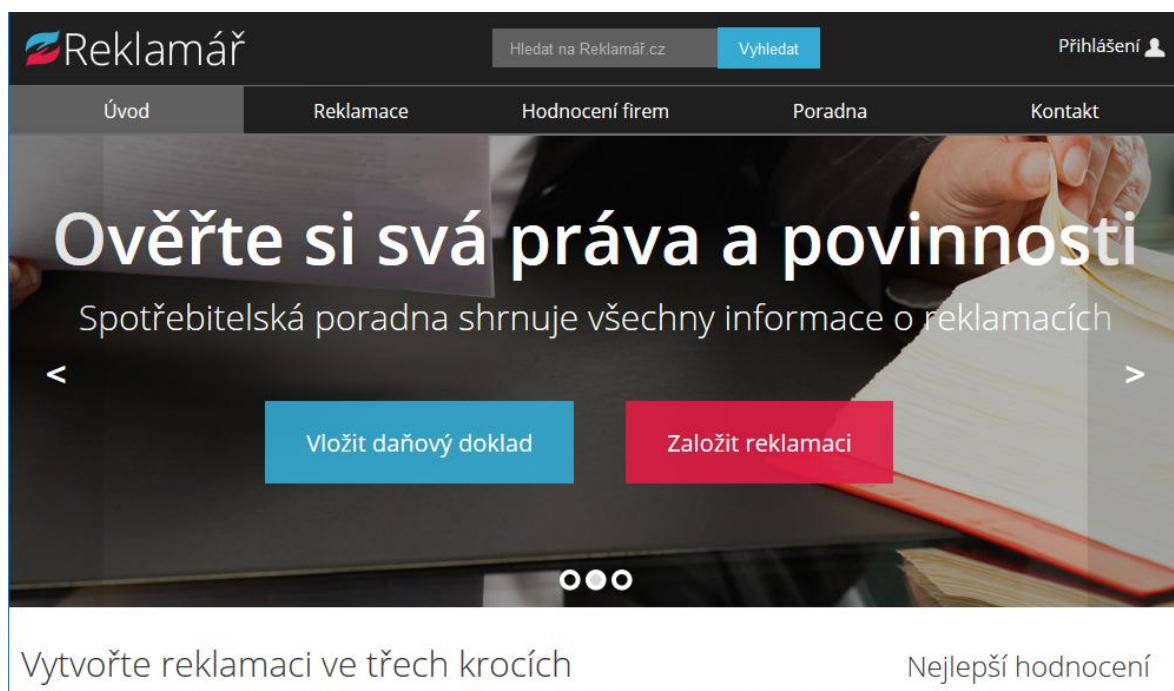
O nás

Webový portál Reklamář je tvořen týmem mladých vývojářů, kteří byly dlouhodobě nespokojeni s neznalostí všeobecného povědomí o reklamacích.

95

Příloha B – Zobrazení úvodní stránky při změně horizontálního rozlišení

1. Rozlišení v rozmezí 800 - 1000px



2. Méně jak 800px



Vytvořte reklamaci ve třech krocích



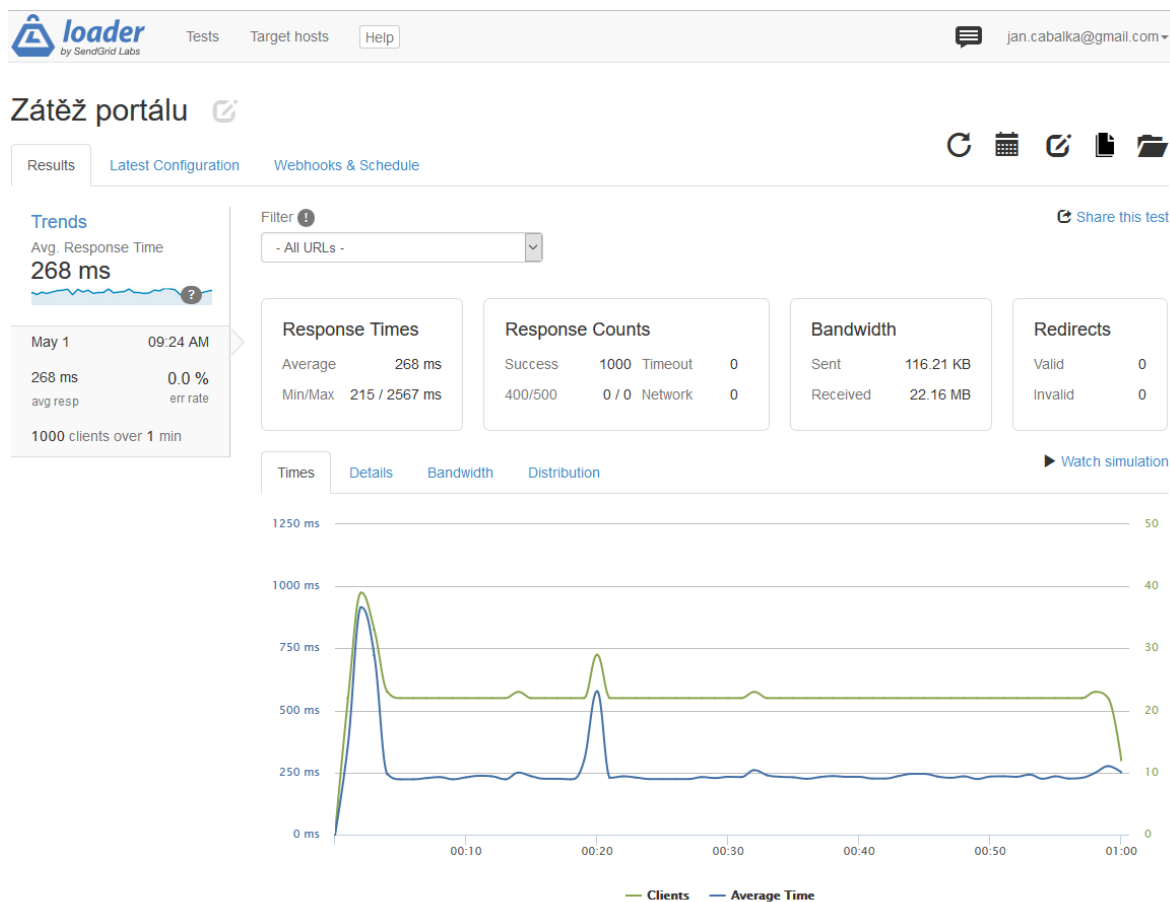
1. Vložení dokladu o nákupu zboží od obchodníka

Doklad v elektronické formě již nemusíte fyzicky archivovat. Reklamář Vás bude informovat o končící době záruky zboží, takže již nikdy neprošvihnete šanci na reklamaci.

Příloha C – Záznamy z testování

1. Zátěž webového portálu 1000 uživatelů v době trvání jedné minuty.

Test byl realizovaný pomocí portálu Loader.io



2. Výkonnostní test webového portálu realizovaný nástrojem Webpagetest.org

WEBPAGETEST

HOME
TEST RESULT
TEST HISTORY
FORUMS
DOCUMENTATION
ABOUT

Web Page Performance Test for www.reklamar.cz/index.php

From: Dulles, VA - Firefox - Cable
1. 5. 2016 9:19:34

A
A
A
F
B
X

First Byte Time
Keep-alive Enabled
Compress Transfer
Compress Images
Cache static content
Effective use of CDN

[Need help improving?](#)
[Raw page data](#) - [Raw object data](#)
[Export HTTP Archive \(har\)](#)
[View Test Log](#)

Summary
Details
Performance Review
Content Breakdown
Domains
Screen Shot

Tester: VM7-IE11-15-192.168.101.195
Test runs: 3

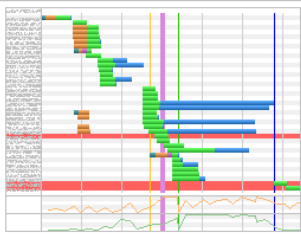

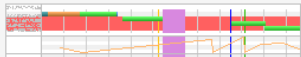

Performance Results (Median Run)

	Load Time	First Byte	Start Render	Speed Index	DOM Elements	Document Complete			Fully Loaded			
						Time	Requests	Bytes In	Time	Requests	Bytes In	Cost
First View (Run 2)	3.251s	0.402s	1.602s	1653	224	3.251s	36	966 KB	3.535s	37	967 KB	\$5---
Repeat View (Run 3)	0.719s	0.339s	0.756s	756	224	0.719s	2	7 KB	1.027s	4	12 KB	

[Plot Full Results](#)

Test Results

Run 1:

	Waterfall	Screen Shot
First View (2.902s)		
Repeat View (0.849s)		

Příloha D – Vybrané stránky portálu

Všechny publikované přílohy jsou nasnímány z testovacího serveru.

1. Hodnocení firem

Úvod	Reklamacie	Hodnocení firem	Poradna	Kontakt
----------------------	----------------------------	---------------------------------	-------------------------	-------------------------

Kategorie

Elektronika




Oblečení a móda

Auto - moto

Sport

Hodnocení firem

TOP 10 firem

-  OKEY s.r.o.
Jedna z největších testovacích firem pospaným v rámci tohoto portálu.
Její podobnost s existující firmou je čistě náhodná
[Detail](#)
-  OKEY s.r.o.
Jedna z největších testovacích firem pospaným v rámci tohoto portálu.
Její podobnost s existující firmou je čistě náhodná
[Detail](#)
-  OKEY s.r.o.
Jedna z největších testovacích firem pospaným v rámci tohoto portálu.
Její podobnost s existující firmou je čistě náhodná
[Detail](#)

2. Poradna

Úvod	Reklamacie	Hodnocení firem	Poradna	Kontakt
----------------------	----------------------------	---------------------------------	-------------------------	-------------------------

Kategorie

[Nákup](#)

[Dokumentace](#)

[Reklamacie](#)

[Zamítnutí](#)

Poradna



- Získejte **užitečné informace** o reklamacích na jednom místě
- **Na co máte při reklamaci právo?** Co může obchodník požadovat?
- **Zajímavosti** o kterých se neví
- **Novinky v legislativě**
- **Zajímavé příspěvky** našich uživatelů

Jaká má spotřebitel práva, objeví-li se vada v záruční lhůtě?

Jedná-li se o vadu, kterou lze odstranit, má kupující právo, aby vada byla odstraněna nebo požadovat výměnu věci. Pokud není takový postup možný, může kupující požadovat přiměřenou slevu z ceny věci nebo od smlouvy odstoupit.

[Zobrazit více informací..](#)

Jaká je délka záruční doby?

Záruční doba trvá:

3. Kontakt

Kontakt

TOMEPOL s.r.o.
Přepeře 162
512 61 Přepeře
Česká Republika
IČO:11111111
Email: info@reklamar.cz

Pomoc s reklamacemi:
Tým Reklamář.cz
Kontaktní e-mail: poradna@reklamar.cz

Technická podpora
Jan Cabalka
Kontaktní e-mail: podpora@reklamar.cz

Kontaktní formulář

Vaše jméno:
Váš e-mail:
Váš telefon:
Kontaktujete:

Text :

Odeslat

4. Přihlašovací formulář

Přihlášení X

Nebo

Váš email:

Vaše heslo:

[Zapomenuté heslo?](#)

[Nemáte účet?](#)

5. Mé doklady

Mé doklady

Zobrazeno záznamů

Hledat:

	Název položky	Firma	Počet kusů	Datum nákupu	Konec záruky	Záruka (měs)	Stav
<input type="button" value="+"/> <input type="button" value="Q"/> <input type="button" value="X"/>	Rychlovarná konvice		1	01.04.2016	01.04.2017	12	Založeno

Zobrazeno 1 až 1 z celkem 1 záznamů

Předchozí Další

6. Založení dokladu

Vložení daňového dokladu

Kde bylo zboží zakoupeno?

Firma:
Nebo **Zadat chybějící firmu**

Informace o nové firmě

Název:
IČ:
Web:
Adresa:
Město:
PSČ:
Stát:

Datum nákupu:

Daňový doklad: No files selected.

Potřebuje pomoci?

- **Vyberte firmu**, od které jste nakoupili. Není-li v seznamu, doplňte údaje o firmě z účtenky pro budoucí kontakt v případě reklamace
- **Datum nákupu je důležitý údaj** - od něho se odvíjí doba záruky
- **Nahrajte doklad** v elektronické formě ve formátu pdf, doc, jpeg nebo jiném. Doklad poté nemusíte fyzicky uchovávat.
- Záruka je standardně **24 měsíců**, někdy ale nabízí prodejce i více

Název položky	Počet kusů	Záruka (měsíců)	Poznámka
<input type="text"/>	<input type="text" value="1"/>	<input type="text" value="12"/>	<input type="text"/>
<input type="button" value="Přidat další položku"/>			
<input type="button" value="Vložit daňový doklad"/>			

7. Administrace – Sekce editor

cAdmin v 1.0 REKLAMÁŘ Jan Cabalka

MENU

- Souhrn
- Editor**
- Uživatelské účty
- Doklady
- Reklamacie
- Firmy
- Hodnocení firem
- Poradna
- Poradna kategorie
- Statistiky
- Administrátorské účty

Editor - poradna.html

hodnoceni_firem.html kontakt.html **poradna.html** registrace.html reklamace.html uvod.html vlozeni_dokladu.html

- Získejte **užitečné informace** o reklamaci na jednom místě
- **Na co máte při reklamaci právo?** Co může obchodník požadovat?
- **Zajímavosti** o kterých se neví
- **Novinky v legislativě**
- Zajímavé **příspěvky** našich uživatelů

body