

Absolvování individuální odborné praxe

Individual Professional Practice in the Company

Zadání bakalářské práce

Student: **Vít Ptošek**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R025 Informatika a výpočetní technika

Téma: **Absolvování individuální odborné praxe**
Individual Professional Practice in the Company

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: banan s.r.o.
2. Struktura závěrečné zprávy:
 - a) Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta.
 - b) Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti.
 - c) Zvolený postup řešení zadaných úkolů.
 - d) Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe.
 - e) Znalosti či dovednosti scházející studentovi v průběhu odborné praxe.
 - f) Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení.

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vede odbornou praxi studenta.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Mgr. Miloš Kudělka, Ph.D.**

Konzultant bakalářské práce: Martin Carbol

Datum zadání: 18.11.2011

Datum odevzdání: 04.05.2012



doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 4. května 2012

.....
Hásek Voj

Rád bych poděkoval vedení společnosti Banan s.r.o. za umožnění absolvování odborné praxe v této firmě. Jmenovitě bych chtěl především poděkovat svému konzultantovi Martinu Carbolovi za celkovou spolupráci a cenné odborné informace a rady. Dále mé poděkování směřuje k mému vedoucímu bakalářské práce Mgr. Milošovi Kudělkovi, Ph.D, jehož rady a připomínky mi pomáhaly při tvorbě této zprávy. V neposlední řadě děkuji své rodině a přátelům za celkovou podporu při studiu.

Abstrakt

Tato bakalářská práce by měla především reflektovat mé pracovní nasazení a působení ve společnosti Banan, sídlící v Ostravě, ve které proběhlo absolvování odborné praxe. Během práce v této firmě jsem plnil povinnosti převážně z oblasti vývoje a správy aplikací a informačních systémů, oboje zpravidla interního typu. Během plnění primárních úkolů se mi dostalo možnosti sekundárně pracovat v sekci technické podpory, kde jsem byl postupně zařazen do pozice experta. Ve své zprávě popisuji nejen charakteristiku společnosti a svého zařazení, ale i postup řešení zadaných úkolů spolu s hodnocením osobního přínosu. Výsledkem mého působení ve firmě měly být především funkční aplikace, jež by firma pravidelně používala, aktivní řešení dílčích problémů na straně klientů společnosti i vnitřních záležitostí. Podílel jsem se také na konzultacích s dalšími kolegy, které vedly k vyřešení problémů i z oblasti ostatních oddělení ve firmě.

Klíčová slova: praxe, PHP, MVC model, databáze, MySQL, SQLite, Informační systém, AJAX, JQuery, DopisOnline, Banan, specialista technické podpory

Abstract

This bachelor thesis reflects on my professional work experience while employed by the Banan Company headquartered in Ostrava. I was employed as an Application and Information Systems Development Manager, both usually of internal type. My duties also included technical support as a secondary responsibility. While holding the position of technical support I was promoted to the level of expert. I describe both the characteristic of the company and its classification, my troubleshooting assignments and how I personally benefited and how it influenced my professional career. The result of my evaluation and recommendations should lead to a functional approach to problem solutions that are beneficial to the company's internal procedures and its clients. I worked closely with my colleagues at Banan while developing problem solving approaches throughout the company.

Keywords: work experience, PHP, MVC model, database, MySQL, SQLite, Information system, AJAX, JQuery, LetterOnline, Banan, technical support specialist

Seznam použitých zkratk a symbolů

AJAX	– Asynchronous JavaScript and XML
CSS	– Cascading Style Sheets - Kaskádové styly
CSV	– Comma-separated values - hodnoty oddělené čárkami
DAIS	– Databázové A Informační Systémy
DDoS	– Distributed Denial of Service - distribuované odmítnutí služby
DNS	– Domain Name System - systém doménových jmen
FTP	– File Transfer Protocol - protokol pro přenos souborů
HTML	– HyperText Markup Language - značkovací jazyk pro hypertext
IM	– Instant Messaging - druh internetové komunikace
IS	– Information system - informační systém
JavaScript	– skriptovací programovací jazyk
MVC	– Model-view-controller - softwarová architektura
OOP	– Object-oriented programming - objekt. orient. programování
PDF	– Portable Document Format - přenosný formát dokumentů
PHP	– Hypertext Preprocessor - skriptovací programovací jazyk
SEO	– Search Engine Optimization - optimalizace pro vyhledávače
SQL	– Structured Query Language - strukturovaný dotazovací jazyk
TZD	– Teorie Zpracování Dat
URL	– Uniform Resource Locator - jednotný lokátor zdrojů
VIA	– Vývoj Internetových Aplikací
VIS	– Vývoj Informačních Systémů
VoIP	– Voice Over Internet Protocol

Obsah

1	Úvod	4
2	Odborné zaměření firmy a pracovní zařazení studenta	5
2.1	Popis profilu společnosti	5
2.2	Popis pracovního zařazení studenta	6
3	Seznam zadaných úkolů	7
3.1	Působení na oddělení technické podpory	7
3.2	Implementace třídy pro DopisOnline	8
3.3	Vývoj a správa IS Faktury	9
3.4	Nová verze aplikace Převod ID	10
3.5	Blokace	12
4	Řešení zadaných úkolů	13
4.1	Technická podpora	13
4.2	DopisOnline	14
4.3	IS Faktury	17
4.4	Převod ID	18
4.5	Blokace	19
5	Získané a scházející znalosti a dovednosti	20
5.1	Teoretické znalosti	20
5.2	Praktické znalosti	21
6	Závěr	22
7	Reference	23

Seznam výpisů zdrojového kódu

- 1 Příklad skriptu u osobního použití zaslání dopisu bez dodejky 16

Seznam obrázků

1	Strukturovaný výpis po úspěšném odeslání.	16
---	---	----

1 Úvod

V době, kdy jsem si měl jako student vybrat téma své bakalářské práce, mě nejvíce oslovila možnost absolvování individuální odborné praxe, ze které bych pak vypracoval zprávu. Přestože jsem narazil na opravdu zajímavá témata prací, praxe pro mne představovala větší přínos pro můj budoucí život z hlediska získání nejen teoretických, ale i praktických zkušeností a dovedností, které bych mohl dále rozvíjet. Také jsem tak mohl zjistit, jakými nedostatky oplývám a jak mé vlastnosti obstojí při práci v týmu. Zjištění, na čem je třeba zapracovat ještě před dokončením studia – před nástupem do zaměstnání, pro mne znamená obrovskou výhodu oproti klasickému vypracování bakalářské práce. Jsem rád, že mé očekávání se vyplnilo a měl jsem tak možnost zkusit si práci v mladém kolektivu rozvíjející se firmy, kde jsem při řešení reálných problémů různého charakteru získal nemálo zkušeností.

Jakmile jsem se takto rozhodl, začal jsem se poohlížet nejprve po firmách uvedených v seznamu, který jsem našel na stránkách fakulty. Bohužel zkušenost byla taková, že se jednatele zastupující konkrétní firmu buď neozvali, nebo se ozvali až za několik měsíců, tedy v době, kdy jsem již vykonával praxi. V jednom případě, kdy o mne byl projeven zájem, mi nabídli pohovor, který se však nemohl uskutečnit, jelikož poslední nabízený termín pohovoru byl v době, kdy jsem byl ve Spojených státech, a zaměstnavateli nevyhovoval můj pozdní přílet. Zkusil jsem si tedy najít společnost mimo tento seznam.

Po domluvě a vzájemné shodě na pracovních podmínkách ohledně práce z IT oboru, jsem byl pozván do společnosti Banan s.r.o. a následně přijat jako stážista na dobu dvou semestrů. Během této doby jsem vykonával zadané úkoly a měl možnost podílet se na vývoji moderní firmy.

V průběhu druhé kapitoly se dále snažím přiblížit vývoj, zaměření a také podnikatelský záměr společnosti. Následně se věnuji popisu své pracovní pozice a požadavkům, které jsem plnil. Třetí kapitola seznamuje s konkrétními úkoly, které mi byly zadány a jejichž řešení s popisem technologií a znalostí, které jsem musel nastudovat a následně praktizovat, rozvádím v kapitole čtvrté. V páté kapitole se zaměřuji nejen na znalosti a dovednosti, které jsem již získal, ale i na ty, na kterých je třeba ještě zapracovat. Závěrečná kapitola podává informace o celkovém průběhu odborné praxe a také se zde věnuji výsledkům v ní dosažených.

2 Odborné zaměření firmy a pracovní zařazení studenta

2.1 Popis profilu společnosti

Společnost Banan (dále jen společnost), která na trhu působí již od roku 2004, se zabývá zejména poskytováním internetových služeb, a to jak placených, tak i neplacených. Jmenovitě se jedná především o webhosting a doménové služby, kde mimo jiné zastupuje roli registrátora české domény. Dále realizuje SEO, webdesign, freehosting, vlastní řešení redakčního i blogovacího systému, e-shopu a miniaplikací pro web. Společnost provozuje také svůj internetový magazín zaměřený na rady a tipy nejen pro zákazníky společnosti.

Celkový počet platících klientů v současné době přesahuje 11 000 zákazníků především z Česka a Slovenska, kdežto bezplatné služby a projekty využívají další desetitisíce. Mezi nejvýznamnější klienty patří například AGENTURA STODOLNI.CZ s.r.o., rádio Helax či Hasičský záchranný sbor Moravskoslezského kraje. V současné době ve společnosti pracuje kolem 30ti lidí v několika sektorech, které bychom mohli z hlediska IT rozdělit následovně:

1. **vývoj a správa software** - Údělem tohoto oddělení je neustálý vývoj a nasazování interního softwaru a softwaru určeného klientům či široké veřejnosti.
2. **technická podpora** - Primárně řeší většinu různorodých problémů a potíží všech klientů. Toto oddělení funguje po celý rok 24 hodin denně, 7 dní v týdnu.
3. **správa systému** - Má za úkol řešit problémy obtížnějšího a závažnějšího charakteru, které vyžadují různá práva a oprávnění. Jedná se především o konfiguraci a správu serverů, aktivních i pasivních síťových prvků, VoIP, databází apod.
4. **webcentrum** - Je tvořeno několika grafiky a kodéry, kteří mají za cíl co nejdříve vytvořit návrh na míru klientům, případně jej dále spravovat. Tento návrh je založen především na redakčním systému či e-shopu společnosti, dále pak například CSS a JavaScriptu.

2.2 Popis pracovního zařazení studenta

V průběhu vykonávání odborné praxe jsem měl to štěstí, že jsem mohl úzce spolupracovat se všemi výše uvedenými odděleními, což mi umožnilo udělat si obrázek o chodu společnosti a získat důležité poznatky při řešení různorodých funkcionalit a problémů.

Před nástupem do praxe jsme se s mým konzultantem Martinem Carbolem domluvili na zaměření úkolů, jakými jsou komunikace a odborná podpora koncových zákazníků, konfigurace sítí (HP, CISCO), konfigurace pobočkové ústředny Asterisk, programování v PHP (s OOP), návrh MySQL databází a správa Linux serverů. Hlavním předmětem mé praxe však měla být především správa a vývoj aplikací a systémů.

Po nástupu do firmy mi byly uděleny potřebné přístupové údaje a oprávnění do systému, různá poučení a seznámení s interními záležitostmi. Po podepsání smlouvy o mlčenlivosti jsem se mohl zapojit do procesu a byly mi zadány první úkoly, na kterých jsem se měl naučit pracovat a sžít se s technologiemi a systémem, což bylo stěžejní pro další práci. Během plnění zadaných úkolů jsem spolupracoval především se svým konzultantem a později i s kolegy z ostatních oddělení, a tak získával nejen zkušenosti potřebné k plnění náročnějších úkolů, ale i zkušenosti s prací v týmu.

3 Seznam zadaných úkolů

Během průběhu odborné praxe jsem měl zadáno hned několik úkolů, které jsem musel splnit. Jednalo se především o vývoj softwaru. Mezitím jsem po celou dobu praxe plnil dlouhodobý úkol, což byla výpomoc v oddělení technické podpory. Dále jsem se podílel na konzultacích a výpomoci různého charakteru i z oblastí sítí, zabezpečení a jiných problémů spojených s ostatními odděleními. Úkoly jsem řešil samostatně i za pomoci zkušenějších spolupracovníků, což mi obohatilo znalosti získané ze školy o dovednosti, které lze získat prací s kolegy z oboru při řešení reálného problému či vzniklé situace.

3.1 Působení na oddělení technické podpory

Na začátku práce bylo potřeba se seznámit s tím, jak to vůbec ve firmě funguje, abych pak mohl na tomto základu stavět. Dobrou příležitostí byla dlouhodobá práce na pozici specialisty technické podpory, kde jsem se dozvěděl mnoho potřebného a navíc jsem se zdokonalil v řešení různých problémů ve stresových podmínkách, s cílem vyřešit je v co nejkratším časovém úseku tak, aby nebyla ochuzena kvalita samotného řešení. Jako přínosu navíc si pak cením zlepšení se v komunikaci s lidmi, která byla pro tuto práci klíčová.

Později jsem byl jmenován do pozice experta a pomáhal jsem se školením a zaučováním nových lidí a řešením problémů na vyšší úrovni. Úzce jsem spolupracoval s kolegy z oddělení správy systému, kde se mi naskytla možnost přechodu.

3.1.1 Technická podpora a vývoj software

Tato práce mi pomáhala také odhalit chyby či nedostatky v IS, aplikacích, skriptech a v celkovém softwaru, který je interně i veřejně hojně využíván, což jsem pak mohl použít jako zadání dalších úkolů v čase stráveném mimo technickou podporu, nebo alespoň navrhnout, jak se případných neduhů zbavit.

3.2 Implementace třídy pro DopisOnline

Když pominu technickou podporu, DopisOnline byl mým prvním úkolem, na kterém jsem měl možnost si vyzkoušet programování v PHP, jelikož jsem do té doby tento skriptovací jazyk nijak nevyhledával a pro podobné účely využíval spíše Python. Důraz byl kladen na OOP, se kterým jsem naopak již nějaké zkušenosti měl.

3.2.1 Služba DopisOnline

DopisOnline je služba České pošty, která má za cíl ulehčit zákazníkům zasílání klasických dopisů. Hlavní myšlenkou je posílání dopisů z pohodlí domova nebo odkudkoliv, kde je přístup k internetu, namísto chování na pobočku České pošty či hledání její poštovní schránky. Funguje to tak, že dopis ve formátu PDF jednoduše zašlete přes určený formulář. Pošta tento dokument přijme, proběhne autorizace uživatele a podle údajů vyplněných ve formuláři daný dokument vytiskne a zašle na příslušnou adresu. Je však nutné, aby si klient založil uživatelský účet, a jeho prostřednictvím mu bude služba dle ceníku vyúčtována

Pro společnost to tedy znamená to, že nikdo ze zaměstnanců již nemusí, například v případě dopisu o expiraci služby, pracně chodit odesílat dopisy na pobočku České pošty. Ovšem i zasílání PDF souborů přes formulář by bylo v případě velkého počtu dopisů celkem pracné a časově náročné, proto se můj konzultant rozhodl využít implementaci této služby do našeho systému.

3.2.2 Zadání

Úkolem tedy bylo vytvořit samostatnou třídu rozhraní sloužícího k automatizovanému podání listovních zásilek elektronickou cestou z aplikací třetích stran, která pak může být použita nezávisle na skriptu, který ji využívá, a to z kteréhokoliv místa interního systému. Jestliže se tedy vygeneruje upomínka na expiraci služby klientovi, který má zaplacené zasílání této upomínky nejen elektronicky, ale i písemně, zavolá se metoda třídy, které se jako parametry předá vše potřebné pro zaslání dopisu pomocí DopisOnline. Dojde tedy automaticky k vygenerování PDF dokumentu, který se zašle na adresu příslušného klienta. Jako zpětná vazba pak poslouží upravený výpis. Ten zahrnuje informace, které je pak možno například zaslat na účetní oddělení jako vyrozumění.

3.3 Vývoj a správa IS Faktury

Další úkol, na kterém jsem se podílel, byl poněkud větší projekt, u něhož jsem opravdu ocenil a zúročil znalosti získané v předmětech TZD, DAIS, VIS a okrajově VIA.

3.3.1 Informační systém Faktury

Každá společnost, která podniká za jakýmkoli účelem, se nejspíše setkává s fakturami dvojího typu. Jsou to faktury vystavené a přijaté. Informační systém, který popisují, se zabývá především fakturami přijatými, jelikož je vhodné mít je uschované a taky nějakým způsobem přehledně založené. Systém má tedy za úkol především evidovat cizí služby, které společnost využívá, a s tím i prostředky na ně vynaložené. Kromě faktur je možno v něm pracovat také s dodavateli, skupinami uživatelů a měnami

3.3.2 Požadavky a funkce

Systém je psán v jazyce PHP s využitím MVC modelu, SQLite databáze a technologie AJAX.

Systém by měl být přehledný a umožňovat:

- přehled faktur, dodavatelů, skupin, měn a logu
- založení nové faktury, dodavatele, skupiny a měny
- editaci a mazání existujících faktur, editaci dodavatelů a skupin
- nastavení upozornění příchodu faktury
- vyhledávání dat dle parametrů
- výpis dat dle nastavitelného řazení a filtrů
- stránkování výpisu
- logování přístupů a změn

3.4 Nová verze aplikace Převod ID

Před mým příchodem do společnosti byla tato funkcionality již naimplementována a nasazena do ostrého provozu. Přestože byla funkční, dalo by se říct, že byla trochu primitivní a časem se od ní začalo očekávat něco víc. Z důvodu každodenního používání, zejména při práci na technické podpoře, se rozhodlo, že je zapotřebí naprogramovat vylepšenou verzi této interní aplikace, což pro mne znamenalo možnost zkusit si opět něco úplně nového. K vývoji aplikace jako je tato, kromě programování skriptů v PHP, byly nutné i širší znalosti ze sektoru správy systému, uplatnění práce ze světa databází a nastavení Cronu.

3.4.1 Aplikace Převod ID

Pro společnost je důležité, aby měla klienty. Ty je nutno nějakým způsobem rozeznávat, což se v databázích řeší určitým jedinečným identifikátorem, tzv. ID. Každý zákazník má tedy své jedinečné ID, které mu je vygenerováno při registraci a pod nímž je pak veden. K tomuto ID jsou přiřazeny údaje klienta, služby, které tento klient využívá, příslušné faktury atd. Každý klient může mít více ID, avšak jedno ID musí patřit právě jednomu klientovi (je tedy pevně dáno). Ovšem občas je třeba s ID klientů poněkud manipulovat. Pokud to řekneme zjednodušeně, tak se jedno ID převádí na jiné. Následující dva konkrétní příklady ukazují, kdy se tohoto převodu využívá a proč.

3.4.1.1 Slévání účtů

Klient Jan Novák vlastní několik domén, ke kterým využívá webhosting, alias a někdy i jiné služby. Těchto domén má u společnosti dohromady 10. Pod svým ID 1 má vedeny čtyři domény, a zbylých šest domén se váže na ID 2, které je rovněž jeho. Panu Novákovi se moc nelíbí, že má všechny své služby rozděleny mezi dvě ID, jelikož se fakturují zvlášť, a navíc by chtěl vidět veškeré své domény v jedné administraci, která se váže na konkrétní ID. V tomto případě stačí jednoduše požádat o sloučení ID, tedy například ID 2 převedeme pod ID 1. Dojde tedy k tomu, že veškeré aktivní služby (jiné nemá smysl převádět) z ID 2 se přesunou pod ID 1, tudíž pan Novák vidí po přihlášení do administrace patřící ID 1 všechny své služby pohromadě a navíc se již nemusí služby rozdělovat na více faktur.

3.4.1.2 Změna vlastníka či určitých údajů

Paní Jarmila Novotná koupila společnost, nebo se nově stala účetní jednotkou. Jelikož tato společnost hodlá nadále využívat našich služeb, je třeba v závislosti na této situaci změnit fakturační údaje. Vzhledem k tomu, že se nejedná pouze o kontaktní emailový účet, číslo ani ulici, ale o IČ nebo DIČ, není možné z ničeho nic u stejného klienta fakturovat služby jinak. Z hlediska účetnictví by tento zákrok byl nepřijatelný. Řešení je tedy nasnadě. Stačí si zaregistrovat nové zákaznické ID, kde je potřeba vyplnit nové fakturační údaje, kterých se celá modifikace týká. Po převodu služeb z původního ID na nově nabyté ID mohou být tyto služby vedeny fyzicky sice na stejného klienta, ovšem s rozdílným ID spjatým s příslušnými fakturačními údaji. Když se poté pro stejné služby na příští faktuře objeví nově navíc identifikační číslo subjektu s daňovým identifikačním číslem, je to v pořádku.

3.4.2 Požadavky a funkce

Aplikace je napsána v jazyce PHP a pracuje s MySQL databází.

Aplikace umožňuje zaměstnancům jednoduchý převod ID klientů. Oproti starší verzi se uživateli po zadání původního ID zobrazí seznam domén, ke kterým se váže alespoň jedna aktivní služba z původního ID. Dále je možno vybrat jednu a více domén z tohoto seznamu, pro které bude převod proveden.

Vše by tedy mělo fungovat následovně:

- zaměstnanec zadá původní ID
- zobrazí se seznam dostupných domén
- po vybrání jedné či více domén je možno zapsat nové ID
- po zapsání existujícího nového ID lze potvrdit převod
- skript se spustí automaticky v pravidelném časovém intervalu
- vše se loguje

3.5 Blokace

Poslední prací, kterou bych zde rád uvedl, je implementace interní aplikace Blokace. Jedná se o řešení blokáci webhostingových služeb klientů přes uživatelské rozhraní namísto ručního spouštění skriptů a správy v linuxovém prostředí.

3.5.1 Aplikace Blokace

V každé společnosti se může lehce stát, že nějaký klient poruší veřejné obchodní podmínky, nebo například koná v rozporu s vnitřním řádem firmy či dokonce se zákonem. U klientů, kteří využívají webhostingové služby, k tomu může docházet například tak, že sdílejí nepatřičný obsah, využívají skripty, které vedou ke spamování, nebo mají co dočinění s jiným škodlivým softwarem. Mohou se však dostat i do opačné role, kdy jsou obětmi DDoS útoků apod. V takových případech je potřeba určitým způsobem odstavit jejich webovou prezentaci. Jelikož by bylo nekorektní jejich data přímo mazat, společnost web či jeho příslušnou část (dočasně) zablokuje, dokud s klientem nedojde k řešení nápravy.

V případě větší četnosti výskytů situací, které by nutně vedly k takovému zákroku, by byla práce s blokacemi příliš pracná a náročná (nikoliv jen z časového hlediska). Ve společnosti tak došlo k rozhodnutí provádět tyto úkony efektivněji, a to za pomoci speciální interní aplikace.

3.5.2 Požadavky a funkce

Aplikace funguje na podobném principu jako Převod ID, používá stejné prostředky i technologie. Liší se v podstatě ve finálním kroku, kdy se provádí rozdílné operace a funkce.

Aplikace slouží kompetentním zaměstnancům pro manipulaci hostovaného prostoru na serveru, který je přidělen klientovi. Konkrétní část webového prostoru či celý web je možno blokovat, pokud je k tomu důvod.

Možnosti aplikace a práce s ní je zamýšlena takto:

- uživatel vidí seznam blokáci s informacemi a grafickým rozdělením dle stavu
- je možno přidat nebo změnit blokaci či odblokovat doménu úplně
- možnost blokace http, poštovního a ftp účtu
- přehled záznamů obsahuje různé podrobné informace
- ověření vstupu a autorizace blokace
- automatické spuštění skriptu v pravidelném časovém intervalu a vyrozumění klienta
- vše se loguje

4 Řešení zadaných úkolů

4.1 Technická podpora

Práce specialisty technické podpory se mi jevila různorodá a přínosná. Za celou dobu jsem se setkal s opravdu velkým množstvím typů problémů i jejich následným řešením. Stejně tak lidé, kterým jsem pomáhal problémy řešit, byli rozdílní. Ovšem práce to byla zajímavá a neméně kreativní.

Řešení problémů se současně odvíjelo za pomoci různých komunikačních metod. Klienci mohli využít telefonní hovor, zákaznický chat, IM klienta, emailovou komunikaci nebo i fórum společnosti. Během komunikace s klienty je třeba stále pozorovat zátěž a výpadky serverů a případně je okamžitě řešit, nebo na ně alespoň upozornit.

Pokaždé jsem se snažil vcítit do role zákazníka a zbytečně jej nezatěžovat složitostmi či vůbec věcmi z oblasti informatiky, jelikož tito klienti byli málokdy z oboru. Většina problémů byla patrná z chybových hlášek nebo laického popisu potíží a řešení tak na sebe nenechalo dlouho čekat. Klientům jsem klasicky pomáhal s doménami, což se týkalo zejména registrací, převodů či jejich správy, jako například nastavení jmenných serverů, DNS záznamů, nastavení aliasů, identifikátorů domény či emailových účtů. Mnohé dotazy a problémy se týkaly webhostingu a výzev k platbě. Mezi tuto práci patřila i pomoc klientům s fakturami, platbami a někdy i se správou jejich webových stránek nebo nastavením e-shopu a SEO. Často jsem také řešil těžkosti spojené s FTP či databázemi, právy k souborům a adresářům, nebo například nastavení počtu dotazů do databáze za hodinu (`max_questions` resource) apod.

Při práci jsem většinou používal příkazový řádek či nástroje dostupné na internetu. Spousta věcí se pak díky interním aplikacím a implementacím v systému prováděla snadno. Užitečnou věcí byla vnitřní Wiki [1] či knowledge base, kde bylo mnoho rad a řešení. Časem jsem obojí editoval a přispíval svými články především pro nové kolegy, které jsem následně i pracovně zaučoval.

4.2 DopisOnline

Jakmile jsem si nastudoval různé tutoriály a literaturu [2] [3] zabývající se programováním v PHP, bylo ještě třeba pořádně porozumět DopisOnline API [4]. Prospěšná byla i webová prezentace sloužící jako centrální referenční bod [5] pro uživatele PHP s vlastní Wiki [6]. Pro běh PHP je nutno mít nakonfigurovaný Apache server, proto jsem zvolil aplikaci VertrigoServ, která mi toto umožnila, a po nějakém čase jsem začal pracovat na implementaci třídy dle požadavků. Jelikož to byl můj úplně první větší úkol a ještě k tomu řešený pomocí technologie, kterou jsem v té době moc neovládal, neobešel se bez menších komplikací, které jsem však nakonec vyřešil, a fungování skriptu tak odpovídá očekávanému výsledku.

4.2.1 Nastudování API DopisOnline

Základní API obsahovalo asi 8 stran a ze začátku bylo pro mne celkem obtížné se v něm vyznat. Několikrát jsem měl nutkání napsat přímo některému správci této služby, jelikož se mi zdálo nevyhovující. Naštěstí jsem se postupně začal v dokumentu orientovat a později to zúročil.

4.2.2 Použití OOP

Velký důraz při implementaci byl kladen na objektové programování. Vytvořil jsem si tedy třídu tak, aby každý objekt měl přesně požadované atributy, které budou při zasílání dále využívány metodou POST spolu s obsahem zasilky a dalšími parametry důležitými pro autorizaci. URL skriptu, který přijímá zasilky, jsem se rozhodl předávat v konstruktoru z důvodu snadné modifikace v případě nutnosti obměny.

4.2.3 Implementace třídy

Samotné objekty by nebyly moc platné, kdyby se jejich dat nedalo správně využít během komunikace se serverem, tedy při odeslání požadavku a případném přijetí jeho odpovědi. Proto bylo zapotřebí použít metodu POST, a po mém zvážení, i knihovny cURL, které PHP nabízí. Komunikace se serverem probíhá výhradně přes zabezpečený protokol HTTPS. Právě kvůli HTTPS a PROXY jsem musel možnosti metody a zvoleného rozšíření trochu více nastudovat. Když jsem měl vše řádně nastaveno, bylo potřeba ještě z dat správně sestavit text (požadavek), který se bude odesílat. Po následném testování jsem zjistil, že odesílání i přijímání už opravdu funguje tak, jak by mělo, a po menších úpravách se teda tato funkcionalita může nasadit.

Nakonec jsem se rozhodl využít možnosti odpovědi serveru a z dat, která mi vrátil ve formátu CSV, jsem zhotovil vlastní strukturovaný výpis, který však pro úplný a bezproblémový chod není nijak stěžejní, jelikož se chybové hlášky s přesnou formulací problému vypisují zvlášť.

Funkčnost jsem zkoušel tak, že jsem požadavek odesílal na URL určenou pro slepé testovací verze skriptu, která veškerá data po přenosu stornuje, aby nedošlo ke skutečnému zaslání dopisu.

4.2.4 Použití a výpis

Tento skript je možné využívat například jak pro své osobní účely, tak i v jiných společnostech, vzhledem k tomu, že kód, který je psán spíše v univerzálním a jednoduchém stylu, nijak nevyužívá k chodu jinou třídu, ani není vázán na konkrétní specifika použití. Skript je navíc možné použít pro zásilky s dodejkou i pro jejich ostatní typy.

Za účelem plné automatizace by se měl skript spouštět s argumenty samostatně předávanými pomocí proměnných ve třídě, kde je plánováno využívání tohoto skriptu. To znamená, že v případě potřeby užití se po správné implementaci provede vše automaticky. Skript je však možno spustit i mechanicky s ručně zadanými parametry, což bude mít za následek stejný proces, s rozdílem osobní obsluhy.

4.2.4.1 Případ automatického použití při expiraci

V třídě, kde se zjišťují expirace klientů, právě dojde k upozornění na konkrétní expiraci. Klient, kterého se to týká, má zapláceno zasílání klasického dopisu. Použijí se tedy jeho data, jakými je například adresa a vygenerovaný PDF soubor obsahující příslušný dopis o expiraci, vše potřebné se předá pomocí proměnných jako parametry a po autorizaci účtu (loginu a hesla) se kompletní požadavek odešle na server, který vrátí zprávu o výsledku. Případně se pak zašle informace o stavu na e-mail.

4.2.4.2 Příklad ručního použití při osobním dopisu

Potřebuji urychleně zaslat dopis, ale z nějakého důvodu nemohu, nebo se mi nechce na poštu. Stačí ručně vyplnit argumenty, tedy adresu, cestu k PDF souboru, druh tisku atd. Připojím registrovaný login a heslo k příslušnému účtu, poté už jen stačí spustit skript a dopis je zaslán. Následně si mohu přečíst vyrozumění o úspěšně dokončené operaci.

Tento způsob je sice pomalejší než při plně automatizovaném použití, ale hodí se pro jednorázové účely, kdy vše proběhne za podstatně kratší dobu než u klasického podání zásilky na poštu a v některých případech i při využití jejich online formuláře.

```
//Nacházíme se v hlavní třídě skriptu
...
//Vytvoření objektu a volání metody třídy s výpisem

$dopis = new DopisOnline("https://online3.postservis.cz/cmdedopis/dopisonline.php");
$vypis = $dopis
    -> Zasilka("uzivatel", "heslo", "0", "195", "0", "csv", "cesta\cenik.pdf", "", "", "", "", "",
        "", "", "", "", "", "", "");

// Po navázání spojení a komunikaci se serverem je možno
// vytisknutí původního i strukturovaného výpisu na obrazovku
echo $vypis;
...
```

Výpis 1: Příklad skriptu u osobního použití zaslání dopisu bez dodejky

```
cenik.pdf;4;20120413086016;neni;2012-04-13;370 20;18.00 Kč;0;195.
```

Upravený výstup

Soubor:	cenik.pdf
Počet stránek:	4
Kód objednávky:	20120413086016
Podací číslo:	neni
Datum podání:	2012-04-13
Podací pošta:	370 20
Cena:	18.00 Kč
Barva tisku:	0 (Černobíle)
Typ výplatného:	195 (Obvyčejně)

Obrázek 1: Strukturovaný výpis po úspěšném odeslání.

4.3 IS Faktury

V době, kdy jsem pracoval na tomto informačním systému, jsem měl již osvojeny alespoň nějaké základy PHP, ovšem zdaleka ne v potřebném rozsahu. Bylo třeba pracovat s více třídami a dohledat si různé funkce, které PHP nabízí, jelikož se mi několikrát stalo, že jsem zbytečně složitě programoval něco, co již bylo naimplementováno v základu.

Jelikož se systém, který využívá více tříd, může časem stát nepřehledným, myslím, že použití architektury MVC bylo správné. S MVC jsem se již v minulosti setkal v jiných programovacích jazycích a technologiích, především v předmětu VIS, takže jsem předpokládal, že použití v PHP pro mne nebude příliš velkou překážkou. Téměř novinkou pro mne však bylo použití AJAXu a JQuery v takovém rozsahu, i když jsem si již něco málo zkusil v předmětu VIA. A konečně přišla na řadu také práce s databází.

4.3.1 Návrh databáze

Podle mého názoru spočívá základ informačního systému ve správně navrženém schématu databáze. Proto jsem se při návrhu snažil zapojit svou intuici spolu se znalostmi získané v předmětu TZD. Po vyhotovení konceptu se nakonec ukázalo, že byl víceméně vyhovující. Jakmile jsem měl toto schéma vytvořené, stačilo jen napsat pár klasických skriptů na vytvoření tabulek, definování dat, omezení a klíčů, případně si předpřipravit dotazy, což jsem měl celkem čerstvě zažito z DAIS. Databáze jako taková byla téměř hotová. Scházela však hlavně data, jejichž manipulace byla primárně zamýšlena přes IS, což byl další krok.

4.3.2 Návrh architektury

Jak už jsem zmínil, zvolená architektura byla MVC. Následovalo tedy rozřazení stránek do sekcí a podsekcí, které budou v IS figurovat, což představuje view. V těchto třídách se většinou vyskytuje pouze HTML kód, který se má zobrazovat, a případně nějaká specifická funkce. Kvůli přehlednosti byly veškeré třídy rozděleny do příslušných adresářů. Stejně tak jsem oddělil grafiku, obrázky atd.

Pro každou sekci je pak v modelu vytvořena třída. Každá z nich obsahuje public funkce, které s ní souvisejí. Klasicky je to funkce pro výpis, editaci, smazání nebo úpravu. Jedná se tedy o funkce, které vracejí určitá požadovaná data, s nimiž se pak dále pracuje. Všechny třídy využívají jednu předdefinovanou třídu určenou pro připojení do databáze a dotazování se. Právě s ní jsem se musel nejdříve seznámit, neboť byla vytvořena již před mým příchodem do společnosti.

V controlleru se klasicky nachází konstruktor a jsou zde zahrnuty i potřebné funkce pro každou sekci, které zajišťují změnu v modelu, jakožto reakci na nějakou událost. Na tomto místě dochází k inicializaci tříd. Posléze objekty manipulují s funkcemi z modelu a využívají dat, která se jim vrátí. Získaná (odchycená) data z modelu se pak opět zobrazují v view. Napsání této třídy tak, aby zabezpečovala bezproblémový chod dle požadavků, zabralo poměrně dost času vzhledem k její důležitosti a rozsáhlosti.

4.3.3 Uživatelské rozhraní

Design je řešen hlavně pomocí HTML a CSS, jímž jsem docílil oddělení struktury stránek od jejich vzhledu. Na některých místech jsem zvolil i PHP vracející HTML kód, což jsem využil například při barevném rozdělení řádků na sudé a liché. Střídáním dvou barev se řádky barevně odliší, což je vyhovující v první řadě kvůli přehlednosti. Kromě vzhledu jako takového bylo potřeba vyřešit i vykreslování a výpis dat v závislosti na výběru. Pro plynulost se mi jevil AJAX jako nejschůdnější řešení, jelikož za pomoci této technologie je možno dosáhnout změny obsahu stránky bez nutnosti znovunačtení obsahu. Ve speciálním dokumentu jsou tak obsaženy funkce napsané v JavaScriptu, které za určitých okolností dokáží změnit cílový obsah. Respektive vrací hodnoty, které vedou ke změně. Taková implementace je celkem náročná. Díky tomu, že jsem použil knihovnu JQuery, to nebylo zas až tak obtížné. Tímto způsobem je docíleno například svižného řazení výpisů dle data vzestupně či sestupně nebo zobrazení tabulky obsahující výpis dat týkajících se upozornění na faktury, a to pouze v závislosti s najetím kurzoru myši na ikonku upozornění.

4.4 Převod ID

K vývoji aplikace pro převod ID bylo stěžejní pochopit, jak systém funguje na pozadí, popř. alespoň to, jak správně ukládat data do databáze, aby bylo zaručeno úspěšné provedení procedur s očekávanými výsledky.

Oproti IS Faktury, skripty této aplikace nepotřebovaly nijak složitou architekturu, v podstatě ani použití objektově orientovaného programování. Stejně tak nároky na vzhled aplikace nebyly příliš vysoké a důraz byl kladen spíše na přehlednost. Naopak bylo opravdu důležité si dávat pozor na různá data, zabezpečení vstupů a ošetření výjimek. Například tam, kde se data vkládají, jaký je jejich formát, jestli odpovídají požadavkům i jiných skriptů, použití funkce `mysql_real_escape_string()` pro případ SQL injection atd.

Dále bylo důležité soustředit se při psaní SQL dotazů sloužící pro výpis dat a manipulaci s nimi. Jako příklad slouží `select` pro výpis všech domén, které patří k zadanému ID a pro něž zároveň platí, že každá z nich má alespoň jednu službu v aktivním stavu. Následně je zapotřebí tato data správně zachytit a předat.

4.5 Blokace

Celá implementace aplikace spočívá, až na malé výjimky, pouze v propojení PHP s MySQL databází. Hlavně prezentuje data a obsluhu skriptů, které probíhají na pozadí. Nejdříve tedy bylo nutné napsat skripty, které se starají o samostatnou funkci blokadí na úrovni správy systému. Tyto skripty jsou psány ve skriptovacím jazyce Perl a bývají využívány v závislosti právě na této aplikaci. Dalším krokem bylo navržení, vytvoření a nastavení odpovídající databáze, která bude naprosto vyhovovat účelům aplikace. Po jakékoliv změně, která je provedena, se nejdříve veškeré operace zaznamenají do databáze a podle těchto záznamů je pak ovlivněn běh blokadí. To vše se děje v pravidelných intervalech díky využití Cronu.

V kódu skriptu se stejně jako v IS Faktury využívá jedna již předem hotová třída sloužící k dotazování v připojené databázi, což mi značně usnadnilo práci poté, co jsem si onu třídu prostudoval a pochopil její zamýšlené použití a možnosti.

Po pochopení logiky skriptů a zvládnutí práce s daty již stačilo zapracovat na grafickém vyjádření a odchyťování dat, která se měla na webových stránkách vypisovat a případně zasílat klientovi pomocí funkce mail().

Při testování bylo nutné vše řádně promyslet, jelikož by mohlo snadno dojít k nějaké transakci, která by měla za následek nechtěné modifikace. V případě, že by snad taková skutečnost nastala a změna se projevila, mohla by se náprava této chyby jen stěží realizovat. Jakmile však byl zaručen požadovaný stav a bezproblémový funkční průběh, mohla se aplikace nasadit do ostrého provozu, díky čemuž je nyní práce s blokacemi mnohonásobně ulehčena. Výhodou také je, že v současné době lze takto zamýšlené úpravy provádět i bez hlubších znalostí Bashe a také je celý proces značně rychlejší.

5 Získané a scházející znalosti a dovednosti

Během vykonávané praxe jsem měl možnost si všimnout a uvědomit, s jakými znalostmi jsem do společnosti přišel, které jsem získal v průběhu práce, a především, do jaké míry jsem schopen je doopravdy využít. Na konci praxe mi pak bylo umožněno zpětně zhodnotit, jakým přínosem pro mě práce ve společnosti byla a čemu jsem se přiučil.

5.1 Teoretické znalosti

Na střední a vysoké škole jsem absolvoval mnoho předmětů. V některých z nich jsem se naučil přemýšlet nad problematikou a řešením trošku jinak, v ostatních jsem se dozvěděl, jak si dohledat příslušné informace, anebo mi v nich byl například poskytnut dobrý základ. Měl jsem možnost seznámit se s hardwarem, programovacími jazyky, funkčností sítí a databází a mnoho jiného. V průběhu praxe jsem pak tento základ mohl prohlubovat a použít hlavně znalosti z předmětů, které jsou spojené s programováním a databázemi a jenž jsem v této práci i zmínil. Teoretické znalosti jsem nepřestal vyhledávat ani během svého působení ve společnosti. Využíval jsem jich zejména v případě, kdy vedly k východisku z daného problému. Dalo by se říct, že jsem využil model řešení, který jsem začal používat při vypracovávání školních projektů. Naučil jsem se také často klást otázku „Jak bychom toto mohli dobře využít?“, případně alespoň nastínit odpověď na ni. I ve firmě byl čas na sebevzdělávání a zlepšování teoretických znalostí z různých okruhů, ovšem vždy s důrazem na jejich využití praxi.

Celkem rychle nyní dokážu v jednotlivých situacích posoudit, zda má smysl věnovat čas práci na určitém zadání zvoleným způsobem, či hledat a nacházet jiný postup nebo návrh a nekomplikovat si tím další práci.

Jistě je ještě spousta teorie, kterou bude v budoucnu třeba prostudovat. Ovšem na rozdíl od školního systému, kdy jsem se něco naučil pro případ, že bych to v budoucnu mohl použít, ve společnosti jsem se konkrétní věc učil, až bylo potřeba ji použít.

5.2 Praktické znalosti

Za celou dobu, kterou jsem ve společnosti strávil, se mi nestalo, že bych se setkal s něčím vysloveně novým, ale v určitém smyslu pro mne byly tyto věci naprostou novinkou. I to, co jsem již nějakým způsobem znal, mi při praktickém využití přinášelo nový rozměr. Najednou jsem pochopil, k čemu je to vlastně dobré, kdy a jak to použít a hlavně proč. Viděl jsem mnoho dílčích částí pracovat dohromady v jednom velkém stroji. Praxe se tedy od teorie nelišila pouze v tom, že při reálném použití se stejná věc může dělat trochu jinak a také se i jinak chová, ale především v tom, že znalosti z různých oborů šly skloubit tak, jak bych ani nečekal. A toto všechno jsem si mohl vyzkoušet na vlastní kůži a nalézt tak odpověď na mou oblíbenou otázku „Co by se stalo, kdyby. . .“. Najednou se mi nedostávalo odpovědí založených na něčem předpokladu, ale na skutečném výsledku. Někdy onen výsledek nebyl zrovna přívětivý a několikrát se mi stalo, že jsem učinil spíše krok zpět, ovšem pro můj osobní přínos to byl krok dopředu.

Ikdyž jsem měl s mnoha technologiemi pramalou zkušenost, někdy dokonce žádnou, tak jsem po nějakém čase dokázal navázat na práci kolegů, nebo alespoň získat dostatečný základ, a dále pak samostatně pracovat. Například během práce s PHP jsem si mimo jiné osvojil i jakési programátorské zvyklosti a návrhové vzory, které mi do té doby nebyly příliš jasné, a neviděl jsem v jejich použití větších výhod.

Největším osobním přínosem však podle mě byla práce v týmu, na kterou se nelze teoreticky připravit. Klíčovým faktorem byla komunikace, tolerance a spolupráce, na čemž jsem chtěl osobně zapracovat. Před každou implementací si nyní rád vyslechnu rady a poznámky zkušenějších kolegů a během řešení se nebojím uznat chybu a opakovaně konzultovat postup.

I během poměrně krátké doby jsem si všiml, že jsem na konci praxe méně chyboval v konkrétních postupech, a dřívější zkušenosti mi napovídaly, čemu se pro příště vyhnout a co by šlo udělat lépe. Stejně jako u teorie, kterou je neustále třeba se učit, u praxe to podle mne platí dvojnásob, jelikož není nutné, abychom dělali chybu, kterou už někdo stačil udělat před námi.

6 Závěr

V této práci jsem chtěl primárně věrně vystihnout své vnímání odborné praxe ve společnosti a shrnout její průběh. Těší mě, že se mi dostalo možnosti být součástí firmy a okusit práci na vývoji softwaru společně s řešením reálných problémů, a to ještě za doby studia. Tato zkušenost mi tak umožnila napravit nedostatky a pomohla mi získat několik nových cenných zkušeností, které se mi do budoucího života budou jistě hodit. V průběhu pracovních dní jsem se setkal s mnoha situacemi a různorodými úkoly, které vedly k osobnímu přínosu nejen v oblasti rozšíření obzoru, zlepšení mých znalostí a dovedností, ale do jisté míry i navýšení sebevědomí a sebejistoty. Svě poznatky a zkušenosti se budu snažit nadále využívat a rozvíjet.

Po mém absolvování praxe je ve společnosti zamýšleno případné navázání na mou práci, jelikož jsou její výsledky poměrně často využívány. Po ukončení jsem podepsal pracovní smlouvu na dobu neurčitou, což pro mne do budoucna znamená, kromě výše popsaných úkonů, šanci podílet se i na podobných, nebo dokonce úplně nových projektech.

Cením si příležitosti vykonání individuální odborné praxe a jejího průběhu. Ačkoli jsem velmi sebekritický, hodnotím své počínání minimálně jako odpovídající předpokladům a s výsledky jsem spokojen. Bohužel jsem tuto práci nemohl v zájmu ochrany know-how rozvést v takovém měřítku, v jakém bych si přál, a uvést více informací, zobrazení a dat, která slouží výhradně pro interní účely společnosti.

Navzdory tomu mě těší, že jsem mohl být během praxe užitečný a má práce měla smysl.

Vít Ptošek

7 Reference

- [1] Banan, *Wikipedie společnosti banan, s.r.o.*, [ONLINE].
URL: <http://wiki.banan.it/>
- [2] LAVIN, Peter, *Object-oriented PHP: concepts, techniques, and code*, No Starch Press, 2006, ISBN 978-159-3270-773.
- [3] ZANDSTRA, Matt, *PHP objects, patterns, and practice. 3rd ed.*, Apress, 2010, ISBN 978-143-0229-254.
- [4] Česká pošta, *Portál Hybridní pošty*, [ONLINE].
URL: <https://online.postservis.cz/>
- [5] PHP, *Main Website*, [ONLINE].
URL: <http://www.php.net/>
- [6] PHP, *The PHP Wiki*, [ONLINE].
URL: <http://wiki.php.net/>
- [7] Banan, *Stránky společnosti banan, s.r.o.*, [ONLINE].
URL: <http://www.banan.cz/>