

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



Bakalářská práce

Převod formátu a tvorba pdf dokumentu pomocí online aplikace

Martin Hlaváček

© 2021 ČZU v Praze

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Martin Hlaváček

Informatika

Název práce

Převod formátu a tvorba pdf dokumentu pomocí online aplikace

Název anglicky

Format converter and making pdf document with online application

Cíle práce

Práce je zaměřena na převod textových dat do formátu, který je pro uživatele dobře čitelný. Práce reflektuje možné metody, historický vývoj těchto technologií a úskalí jejich používání včetně omezujících podmínek. Cílem práce je popsat možnosti práce s daty a to jak v teoretické, tak v praktické rovině. Práce se zabývá analýzou vhodných prostředků pro tuto aplikaci. Dále je v práci zpracována rešerše dalších možných řešení daného problému.

Metodika

Pro zpracování této práce jsou použity metody sběru dat, informací a následnou manipulaci s nimi. Zdrojem informací pro teoretickou část jsou odborné články, knihy a další zdroje vztahující se k této problematice. Praktická část je ukázkou ve formě funkčního online programu, který používá dané metody a postupy pro převedení textových dat na formát, který je pro člověka dobře čitelný a připravený k tisku.

Doporučený rozsah práce

30 – 40 stran

Klíčová slova

PDF, xls, XLS, aplikace, tvorba PDF, Převodník PDF, Práce s daty

Doporučené zdroje informací

DARIE, C. *AJAX a PHP : tvoříme interaktivní webové aplikace profesionálně*. Brno: Zoner Press, 2006. ISBN 80-86815-47-1.

DENNIS, A. *Tvorba PDF pomocí Adobe Acrobat : průvodce pro profesionály v DTP a prepressu*. Brno: Computer Press, 2003. ISBN 80-7226-718-3.

DOMES, M. *Tvorba internetových stránek pomocí HTML, CSS a JavaScriptu*. Kralice: Computer Media, 2005. ISBN 80-86686-39-6.

GILMORE, W J. – POKORNÝ, J. *Velká kniha PHP 5 a MySQL : kompendium znalostí pro začátečníky i profesionály*. Brno: Zoner Press, 2011. ISBN 978-80-7413-163-9.

POKORNÝ, M. *PHP nejen pro začátečníky*. Kralice: Computer Media, 2005. ISBN 80-86686-38-8.

Předběžný termín obhajoby

2020/21 LS – PEF

Vedoucí práce

Ing. Václav Lohr, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 2. 11. 2020

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 5. 11. 2020

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 02. 01. 2021

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci „Převod formátu a tvorba pdf dokumentu pomocí online aplikace“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 15.března 2021_____

Poděkování

Chtěl bych zde poděkovat Ing. Václavu Lohrovi, Ph.D., za odborné vedení, připomínky a užitečná doporučení.

Převod formátu a tvorba pdf dokumentu pomocí online aplikace

Abstrakt

Tato bakalářská práce se zabývá analýzou vhodných řešení pro převod textových dat do formátu, který je pro uživatele snadno čitelný. Práce reflektuje možné metody online konverze, jejich úskalí a historický vývoj včetně srovnání těchto metod. Zjištěné poznatky jsou dále uplatněny ve tvorbě vlastní webové aplikace, jež tyto technologie využívá a exportuje data ve formátu pdf.

Klíčová slova: pdf, xlsx, xls, aplikace, tvorba pdf, převodník pdf, práce s daty

Format converter and making pdf document with online application

Abstract

This bachelor thesis analyses suitable solutions for converting data to human readable format. Thesis analyses and compares online converting solutions, their disadvantages, security breaches and their evolution. Based on these findings, the online converter application has been made to export data to pdf format.

Keywords: pdf, xlsx, xls, app, create pdf, pdf converter, work with data

Obsah

1. Úvod.....	12
2. Cíl práce a metodika	13
2.1 Cíl práce	13
2.2 Metodika	13
3. Teoretická východiska	14
3.1 Databázový systém.....	14
3.2 Relační databáze.....	14
3.3 FTP.....	15
3.4 Internet	15
3.5 Síť WWW a protokol HTTP	16
3.6 Statické www stránky	17
3.7 Shoptet.....	17
3.8 Dynamické webové aplikace.....	17
3.9 Formát pdf a jeho generování z jiných formátů	18
3.10 Framework	18
3.11 MVC Framework	18
3.12 MVC komponenty.....	19
3.12.1 Model.....	19
3.12.2 View - pohled.....	19
3.12.3 Controller - řadič.....	19
4. Prostředky tvorby dynamické webové aplikace a vybrané cloudové aplikace ..	20
4.1 Porovnání programovacích jazyků.....	20
4.1.1 PHP	20
4.1.2 JavaScript.....	21
4.1.3 Zhodnocení	21
4.2 Řešení konverze pomocí cloudových aplikací třetích stran.....	22
4.2.1 Služba Google Docs.....	22
4.2.1.1 Konverze dokumentů pomocí Google Drive.....	22
4.2.1.2 Konverze dokumentů za použití tagů v Google Docs – tvorba masky25	
4.2.1.3 Import dat	26
4.2.1.4 Nastavení aplikace Document Data Studio	27
4.2.1.5 Možnost zaslání výstupního souboru do e-mailu	27
4.2.1.6 Uložení souborů na Google Drive.....	28
4.2.1.7 Spuštění generování dokumentu.....	28

4.2.2	Řešení JotForm	28
4.2.2.1	Tvorba JotForm formuláře	28
5.	Vlastní práce	35
5.1	Představení podniku	35
5.2	Představení problému a jeho možného řešení	35
5.3	Trend momentálního vývoje	36
5.4	Představení aplikace	36
5.4.1	Specifikace požadavků	37
5.4.2	Vývojář	37
5.4.3	Koncový uživatel neboli návštěvník	37
5.4.4	Bezpečnost	38
5.4.5	Adresářová struktura	38
5.4.6	Vstupní data a obsluha aplikace	39
5.4.7	Výstup z aplikace	41
5.4.8	Předpoklady pro spuštění aplikace	42
5.4.9	Popis přiložených souborů	42
6.	Závěr	42
7.	Seznam použitých zdrojů	43
8.	Přílohy	46
8.1	Příloha 1 - Struktura kódu, část /index.php	46
8.2	Příloha 2 - Struktura kódu, část /create_pdf.php	48
8.3	Příloha 3 - Struktura kódu, část Classes/config.php	56
8.4	Příloha 4 - Struktura kódu, část Classes/tcpdf.php	58

Seznam obrázků

Obrázek 1 - Nákres komponent MVC	19
Obrázek 2 – Tvorba vstupní složky	23
Obrázek 3 – Nahrání vstupních souborů.....	24
Obrázek 4 – Výběr výstupního formátu	24
Obrázek 5 – Přidání tagů pro další práci s proměnnými.....	25
Obrázek 6 – Párování tagů a dat	26
Obrázek 7 – Tvorba šablony e-mailu.....	27
Obrázek 8 – Vytvoření JotForm šablony a přidání možnosti exportu	29
Obrázek 9 – Přidání doplňku pro export pdf dokumentu	29
Obrázek 10 – Aktivace doplňku pro export.....	30
Obrázek 11 – Úprava šablony s vloženými tagy	31
Obrázek 12 - Přiřazení dat ze šablony k tagům	31
Obrázek 13 - Přiřazení tagů ze šablony k tagům v datech doplňku.....	32
Obrázek 14 – Pojmenování výstupního souboru	33
Obrázek 15 – Uložení exportovaného souboru do adresáře	33
Obrázek 16 – Export a zaslání exportovaného souboru na e-mail	34
Obrázek 17 – Vstupní data.....	39
Obrázek 18- Frontend aplikace.....	40
Obrázek 19- Výstup z aplikace	41

1. Úvod

Hlavním důvodem pro vznik této práce je potřeba řešit situaci, kdy data jsou systémem předkládána v textovém řetězci, ve kterém jednotlivé sekvence oddělují čárky, a jejich převod na formát, jenž bude přehlednější pro následné zpracování člověkem. Jako vhodná metoda se jeví interpretace a rozlišení dat pomocí jejich lepšího rozmístění na jednu stránku A4 a vytištění této sestavy. Pro snazší rozlišení dat v textovém řetězci je vhodné využít zvýraznění určitých dat na základě podmínek, jejich jednotlivé rozlišení pak umožňuje menší chybovost při následném lidském zpracování. Díky lepšímu propojení databáze objednávek a expedice je možné výrazně zrychlit kompletaci objednávek při menším počtu chyb, a tím snížit náklady. V práci je uvedeno více možných řešení daného problému včetně cloudového řešení. Tato řešení jsou porovnána, i s ohledem na možná bezpečnostní rizika. Výhody tisku expedičních listů oproti elektronické kompletaci jsou především v možnosti bezproblémového zavedení nové aplikace a paralelního použití současného systému bez nutnosti zaškolení personálu a výrazných změn procesů firmy, které by znamenaly zvýšené náklady. Tato práce popisuje fungování aplikace a jejích potřebných komponent a následně její integraci do fungování podniku, kde je vyzkoušena v praxi. Pomocí ukázkové aplikace jsou data oddělená čárkou převáděna do čitelné podoby pdf formátu, který je ideální k vytištění.

2. Cíl práce a metodika

2.1 Cíl práce

Práce je zaměřena na převod textových dat do formátu, který je pro uživatele dobře čitelný. Práce reflektuje možné metody, historický vývoj těchto technologií a úskalí jejich používání včetně jejich nevýhod a omezení. Cílem práce je popsat možnosti práce s daty, a to jak v teoretické, tak v praktické rovině. Práce se zabývá analýzou vhodných prostředků pro tuto aplikaci. Dále práce obsahuje porovnání dalších možných řešení daného problému.

2.2 Metodika

Pro zpracování této práce jsou použity metody sběru informací a dat a následná manipulace s nimi. Zdrojem informací pro teoretickou část jsou odborné články, knihy a další zdroje vztahující se k této problematice. Praktická část je ve formě ukázky funkční online aplikace, jež používá dané metody a postupy pro převedení textových dat na formát, který je pro člověka dobře čitelný a připravený k tisku.

3. Teoretická východiska

Teoretická východiska v této bakalářské práci jsou pojmy, které jsou klíčové pro pochopení tématu práce a její problematiky. Následně bude uvedeno členění a popis jednotlivých systémů.

3.1 Databázový systém

Databáze (DB) je soubor dat, tedy údaje nebo data vztahující se k určitému problému. Jako příklad použití databáze lze uvést Zlaté stránky.

Každou databázi je třeba řídit tak, aby nedošlo k neoprávněné manipulaci s databází nebo k jejímu poškození. Dnes se již používá výlučně databázové zpracování, nikoliv agendové, které má vícero nedostatků. Mezi nejvýznamnější patří obtížně zajistitelná integrita dat, složité zabezpečení systému a ochrana dat proti napadení a problémy s narůstající redundancí dat, která způsobuje jejich nekonzistenci. Databázové zpracování umožňuje přístup k datům pouze skrze systém řízení báze dat DBMS, přímá manipulaci s daty není možná. Uživatel tak může přistupovat k datům pomocí aplikačních programů, ale zároveň nenaruší fyzická data v databázovém systému. Bezpečný chod DBS zabezpečuje systém řízení báze dat (DBMS), jenž zajišťuje řízení přístupu uživatele k datům a administraci dat na úrovni jejich fyzického uložení, a uživatelé komunikují právě s ním. Databáze (DB) a DBMS spolu tvoří DBS, neboli databázový systém. Jednotlivá elementární data (údaje) jsou v databázi nazývána datová základna. (DOBEŠOVÁ, 2004)

3.2 Relační databáze

Vstupem pro aplikaci v této práci jsou elementární data, z nichž je nutné vytvořit databázi tak, aby byla možná další práce s daty. „Relační databáze je typ databáze, který ukládá vzájemně propojené datové body a poskytuje k nim přístup. Relační databáze jsou založeny na relačním modelu, intuitivním, přímočarém způsobu vyjádření dat v tabulkách. V relační databázi je každý řádek v tabulce záznam s jedinečným ID nazvaným klíč. Sloupce tabulky obsahují atributy dat a každý záznam má obvykle hodnotu pro každý atribut, což usnadňuje navázání vztahů mezi datovými body.“ (ORACLE, 2017)

3.3 FTP

Jedná se o protokol, který zprostředkovává přenos souborů mezi počítači pomocí počítačové sítě (internetu). Umožňuje rychlé nahrání souborů na server, jeho zabezpečení však není dostatečné. (FOROUZAN, 2009) Pro lepší zabezpečení přenášených souborů se protokol používá v kombinaci s SSH protokolem nebo přímo ve specifikaci protokolu FTPS. Hesla a uživatelská jména jsou v FTP protokolu bez dalšího zabezpečení přenášena jako prostý text bez šifrování, nejedná se tak o vhodný způsob přenášení citlivých dat. (Nurdletech, 2014)

3.4 Internet

Internet vznikl z původně armádního projektu ARPANET, kdy v šedesátých letech minulého století americká armáda hledala způsob, jakým zajistit komunikaci armádních počítačů, které se nacházely v různých městech nebo státech. Síť byla již v té době decentralizovaná, a to pro její ochranu v případě zničení některého z uzlů. (DARPA, 2007)

Internet je celosvětové propojení počítačů pomocí rodiny protokolů TCP/IP. Protokoly TCP/IP se dělí do čtyř vrstev. Aplikační vrstva je zastoupena např. FTP protokolem pro přenos souborů, POP3 protokolem pro přenos pošty, HTTP protokolem viz kapitola 3.5 nebo SSH protokolem pro šifrovaný přenos dat. Transportní vrstva se skládá z protokolu TCP, který je spolehlivější, datově náročnější a zajišťuje integritu dat, a UDP protokolu, který je rychlejší, ale nekontroluje datovou integritu. Síťová vrstva obsahuje např. IP protokol zajišťující přenos dat na jednotlivá zařízení, ICMP protokol používaný pro chybové zprávy nebo DHCP protokol přidělující IP adresy jednotlivým zařízením. Vrstva síťového rozhraní zprostředkovává vlastní přenos pomocí drátových a bezdrátových technologií Ethernet, Wifi. (IETF RFC, 1974) K tomuto modelu existuje alternativní referenční model ISO/OSI, jenž má sedm vrstev: aplikační vrstva, prezentační vrstva, relační vrstva, transportní vrstva, síťová vrstva, linková vrstva, fyzická vrstva. (PETERKA, 1992)

3.5 Síť WWW a protokol HTTP

WWW je zkratka pro Word Wide Web. Síť WWW se skládá z následujících komponent:

1. Architektura Klient/Server – uživatel zasílá prostřednictvím URL požadavek na server s informací o tom, na které URL se chce připojit. Server na základě dotazu uživateli předá požadovaná data.
2. Hypertext – odkaz, který uživatele po kliknutí přeměruje na jinou sekci na dané stránce nebo na stránku zcela jinou.
3. Identifikátor zdroje (anglicky resource identifier) – identifikátor nesoucí informaci o umístění serveru a jeho konkrétní stránky. Toto umístění se označuje jako URL.
4. Značkový jazyk (anglicky markup language) – jedná se o jazyk, který obsahuje instrukce pro sazbu textu včetně obrázků a dalšího obsahu. Příkladem značkových jazyků je HTML nebo XML.

HTTP je protokol, který zprostředkovává komunikaci se servery a zajišťuje přenos jednotlivých HTML požadavků z aplikační vrstvy na server. HTTP protokol se poprvé objevil okolo roku 1990. První verze 0.9, určená pro přenos textu, fungovala pomocí metody GET, a to jak pro přenos dat, tak pro přenos chybových zpráv. Pro nemožnost přenosu složitějších dat a souborů vznikla verze 1.0, která zavedla stavový řádek pro odpověď a možnost editovatelné HTTP hlavičky v požadavku i v odpovědi pomocí metody HEAD a POST. (POKORNÝ, 2005) Současná verze protokolu je 1.1, tato verze umožňuje provoz více WWW serverů na stejné doméně a její hlavní výhodou je schopnost udržovat TCP spojení a přenos více souborů po sobě. HTTP 2 je verze protokolu přijatá 14. května 2015 jako standard RFC 7540. Zastoupení serverů, které tuto technologii zvládnou, se neustále zvětšuje. V březnu roku 2021 překročilo 50 %. (W3TECHS, 2021) Hlavním důvodem pro nasazení protokolu HTTP 2 je snížení latence a zvýšení tak rychlosti načítání. HTTP 3 je nově vyvíjená verze protokolu označovaná také „HTTP over QUIC“, která byla publikována jako šablona dne 24. října 2018. Tato verze umožní ještě rychlejší načítání.

3.6 Statické www stránky

Statická www stránka je koncovému uživateli zobrazena přesně tak, jak byla uložena, na rozdíl od dynamické webové aplikace, která je měněna v průběhu času nebo základě uživatelské interakce. Tento typ stránek se již moc nevyužívá, jelikož uživateli neumožní např. vložit komentář nebo jinak se stránkou komunikovat. Tento typ stránek je vhodný jako výstup pro informační stránku. Také je možné jej použít v kombinaci s WYSIWYG (What you see is what you get, česky „co vidíš to dostaneš“) editorem. (DOMES, 2005) Tento editor umožňuje rychlý náhled dat například při tvorbě webového formuláře přes webové aplikace jako je Google Document Studio. Statická webová stránka nabízí vývojářům rychlý náhled výsledné HTML stránky.

3.7 Shoptet

„Shoptet se na českém a slovenském trhu specializuje na pronájem e-shopových řešení, která jsou oblíbená zejména pro jednoduché a intuitivní ovládání. V současné chvíli Shoptet pomáhá podnikat již 25861 aktivním e-shopům, což je třetina všech internetových obchodů působících na českém a slovenském trhu. Také nabízí pokladní systém, který umožňuje správu e-shopu a kamenného obchodu v rámci jednoho on-line systému. Samozřejmostí je vše potřebné pro EET a GDPR.“ (SHOPTET, 2021)

3.8 Dynamické webové aplikace

Pro moderní aplikace jako jsou online hry, e-shop systémy nebo fóra je běžné použití relací (session), kde uživatel na základě svého přihlášení nebo jen používané relace vidí jiný obsah než další uživatelé. Na základě interakce uživatele je spuštěn skript, který reaguje na akci uživatele. Obsah v prohlížeči uživatele je vygenerován nejčastěji pomocí kódu v JavaScriptu nebo PHP. Dynamické webové aplikace je možné připojit na databázový systém, čímž vzniká velmi komplexní aplikace s možností práce s daty. Obecně lze dynamické webové aplikace rozdělit na dvě skupiny: serverové a klientské. Serverové aplikace, nejčastěji psané ve skriptovacích jazycích, generují obsah stránky již na serveru a tento vygenerovaný obsah předávají uživateli na základě jeho dotazu k

zobrazení na jeho koncovém zařízení. Na hardware ani software uživatele nejsou kladeny zvýšené nároky. Mezi serverové dynamické webové aplikace patří například aplikace psané v jazyce PHP. Klientské dynamické webové aplikace musí naopak uživatelské zařízení podporovat, což zvyšuje nároky na hardware a software na straně uživatele, ale ulehčuje zatížení serveru. Příkladem takové aplikace je aplikace psaná v jazyce JavaScript. (DOMES, 2005)

3.9 Formát pdf a jeho generování z jiných formátů

Formát pdf je velmi vhodný formát pro sazbu tisku, jelikož se dá říci, že se na všech zařízeních zobrazuje stejně a podobu exportovaného pdf souboru lze přiblížit obrázku. (DENNIS, 2003) Pro generování pdf z jiných formátů nebo textového řetězce je možné použít mnoho aplikací. Pro potřeby této práce je použita třída TCPDF hlavně z důvodu jejího opensource řešení a kompatibility s jazykem PHP. (TCPDF, 2002)

3.10 Framework

Framework je softwarová struktura, která poskytuje základní rámec k řešení daného problému. Nabízí tak vývojáři možnost zaměřit se při vývoji webové aplikace na konkrétní chod aplikace a nikoliv na přechod mezi jednotlivými stránkami nebo vzhled frontendu aplikace. Ve zkratce nabízí struktura framework možnost, jak pomocí šablon navrhnout různé části aplikace. Pokud je framework z aplikace odinstalován, aplikace přestane fungovat. (TUTORIALSPOINT, 2015)

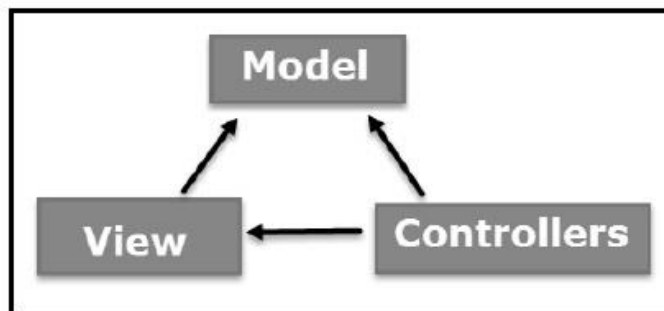
3.11 MVC Framework

Struktura MVC (Model-View-Controller) nabízí základní rámec k řešení struktury aplikace. MVC Framework separuje aplikaci na tři logické celky, a to na model, pohled a řadič. Každý z těchto celků je určen k řízení konkrétní části aplikace. MVC je jeden z nejčastěji používaných průmyslových standardů tvorby aplikací. MVC Framework umožňuje velkou škálovatelnost a rozšiřitelnost aplikace bez velkých zásahů do jejího kódu. (TUTORIALSPOINT, 2015)

3.12 MVC komponenty

Komponenty MVC viz Obrázek 1.

Obrázek 1 - Nákres komponent MVC



Zdroj: MVC Framework - Introduction [online]. [cit. 2021-25-02].

https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm

3.12.1 Model

Tato komponenta reprezentuje veškerou datově-relativní logiku, se kterou uživatel pracuje. Model reprezentuje data, která jsou přenášena mezi pohledem a řadičem, a jakákoliv další data v systému. Model je využíván například v situaci, kdy si uživatel vyžádá data z databáze, změní je a vrátí zpět do databáze. (TUTORIALSPOINT, 2015)

3.12.2 View - pohled

Pohledová komponenta je používána pro uživatelské rozhraní a jeho logiku. Jedná se tedy o vše, co uživatel vidí, jako jsou např. zaškrťovací boxy, textová pole nebo rozbalovací nabídky. Jde o prostředí, se kterým komunikuje a interaguje koncový uživatel. (TUTORIALSPOINT, 2015)

3.12.3 Controller - řadič

Řadič slouží jako komunikační prostředek mezi modelem a pohledem a vyhodnocuje veškerou logiku a příchozí požadavky. Manipuluje s daty za pomoci modelu a spolupracuje s pohledem na tvorbě finálního výstupu pro uživatele. Uživatel zadá data do pohledu a řadič zajistí aktualizaci dat v modelu pro uživatele. Následně bude stejný řadič použit k předání dat do pohledu a ten zobrazí data uživateli. (TUTORIALSPOINT, 2015)

4. Prostředky tvorby dynamické webové aplikace a vybrané cloudové aplikace

Hlavní část této práce se věnuje realizaci dynamické webové aplikace za použití skriptovacího jazyka PHP. Tato kapitola by měla přiblížit prostředky, které se v aplikaci používají, i její alternativu v podobě cloudových aplikací.

4.1 Porovnání programovacích jazyků

4.1.1 PHP

Počátky programovacího jazyka PHP sahají do roku 1994, kdy dánský programátor Rasmus Lerdorf zkombinoval Jazyk C a proměnné z jazyka Pearl a navíc přidal možnost napojení na databázi. (DARIE, 2006) Tato kombinace umožnila tvorbu prvních dynamických webových aplikací, kterou využil na dnes již běžné počítačové přístupy na web. PHP se pak dále významně vyvíjelo v Izraeli. Vývoj jazyka PHP probíhal formou postupného přepisování kompilátoru od první verze až do verze PHP 3. Tento kompilátor se dále zdokonaloval dalším přepisováním jádra, a to až k tvorbě Frameworku Zend Engine.

Výsledkem této snahy byla nová verze PHP 4, která byla dále zlepšena o možnost objektově orientovaného programování, čímž došlo k přechodu na novou verzi PHP 5. (GILMORE, 2011) Verze PHP 6 nebyla nikdy vydána, byla vyvinuta přímo verze PHP 7, jež nabízí vyšší rychlost, nižší paměťovou náročnost a anonymní třídy a funkce.

V listopadu 2020 vyšla nová verze PHP 8.0, která nabízí pojmenované argumenty a atributy spolu s definicí třídních proměnných v konstruktoru. (W3TECHS, 2021) (PHPNET, 2020)

Programovací jazyk PHP je nejrozšířenější (W3TECHS, 2021) skriptovací programovací jazyk určený především k tvorbě dynamických webových stránek a aplikací. Aplikace napsané v jazyce PHP se vyznačují tím, že se provádí na straně serveru. Popularitu získal díky snadnému použití, velkému množství funkcí, možnosti snadného rozšíření a díky komunitě, která jej stále vyvíjí, rozšiřuje a zdokonaluje.

Hlavní výhodou jazyka PHP je přímá specializace na webové stránky a přístup k velkému množství funkcí již v základní knihovně. Další výhodou je možnost přidání dalších

externích knihoven a tím rozšíření funkcí. Je třeba zmínit nativní podporu databázových systémů a dnešní standardizovanost PHP, která umožňuje umístění aplikace na většinu serverů bez větších zásahů. K většině funkcí je dostupná dokumentace a mnoho návodů. Nevýhodou mohou být nekonzistentní názvy funkcí, např. funkce s odlišnou logikou použití jako `strpos()` (uložení změny do proměnné a následné nahrazení stringu) a `str_replace` (přímá náhrada stringu). V některých případech se také vyskytuje nejednotné pořadí parametrů a nejednotné názvosloví skupin funkcí. (Trnka, 2007)

4.1.2 JavaScript

JavaScript je objektově orientovaný skriptovací jazyk, jenž najde využití ve většině moderních dynamických webových stránek, které nabízejí interakci uživatele, např. animace, tlačítka a textová pole. Tento skriptovací jazyk začal být hojně využíván v devadesátých letech 20. století. S jazykem JavaScript přišla možnost rychlého a snadného vývoje dynamických webových aplikací. JavaScript se spouští na straně uživatele po načtení webové stránky, viz kapitola 3.8. Uživatel prostřednictvím JavaScriptu zašle požadavek na server, server pak zašle uživateli zpět HTML stránku a script, který je opět zpracován prohlížečem, což má určitá bezpečnostní rizika, např. možnost spuštění škodlivého kódu ve formě vloženého `iframe` formuláře a odkazu na jinou stránku nebo načítání souboru přímo v prohlížeči. Výhodou JavaScriptu je naopak rychlost načítání stránky hlavně v případě, že dochází ke změnám obsahu uživatelem a server tak nemusí být neustále zatěžován požadavky. Příkladem může být formulář, který uživatele upozorní na chybějící číslo popisné u vkládané adresy. Další výhodou JavaScriptu je možnost snadné přizpůsobitelnosti webové stránky přímo jednotlivým zařízením a velikosti jejich obrazovek. Programy napsané v JavaScriptu se umísťují přímo do validního HTML kódu a je tak velmi snadné s JavaScriptem pracovat. (DARIE, 2006)

4.1.3 Zhodnocení

Využití jazyka PHP je ideální řešení pro méně rozsáhlé dynamické webové aplikace, u rozsáhlejších je jeho limitujícím faktorem rychlost zpracování. Výhodou je jeho všeobecné použití a tedy jednoduchost hledání hostingu. S jazykem PHP je možné pokrýt většinu

běžných požadavků, a to za použití obvykle volně dostupných knihoven bez dalších poplatků. Existují další alternativy, z nichž nejznámější jsou interpretovací jazyky Go či Python. Jejich použití je ale nákladnější a vyžaduje většinou běh virtuálního serveru, odmění se však vyšším výkonem a stabilitou aplikace.

4.2 Řešení konverze pomocí cloudových aplikací třetích stran

Konverzi dokumentů je možné provést pomocí online cloudových aplikací třetích stran. V této práci jsou představena dvě vybraná řešení, a to služba Google Docs ve spojení s aplikacemi Google Data Studio, Google Drive a Google Documents a její alternativu JotForm.

4.2.1 Služba Google Docs

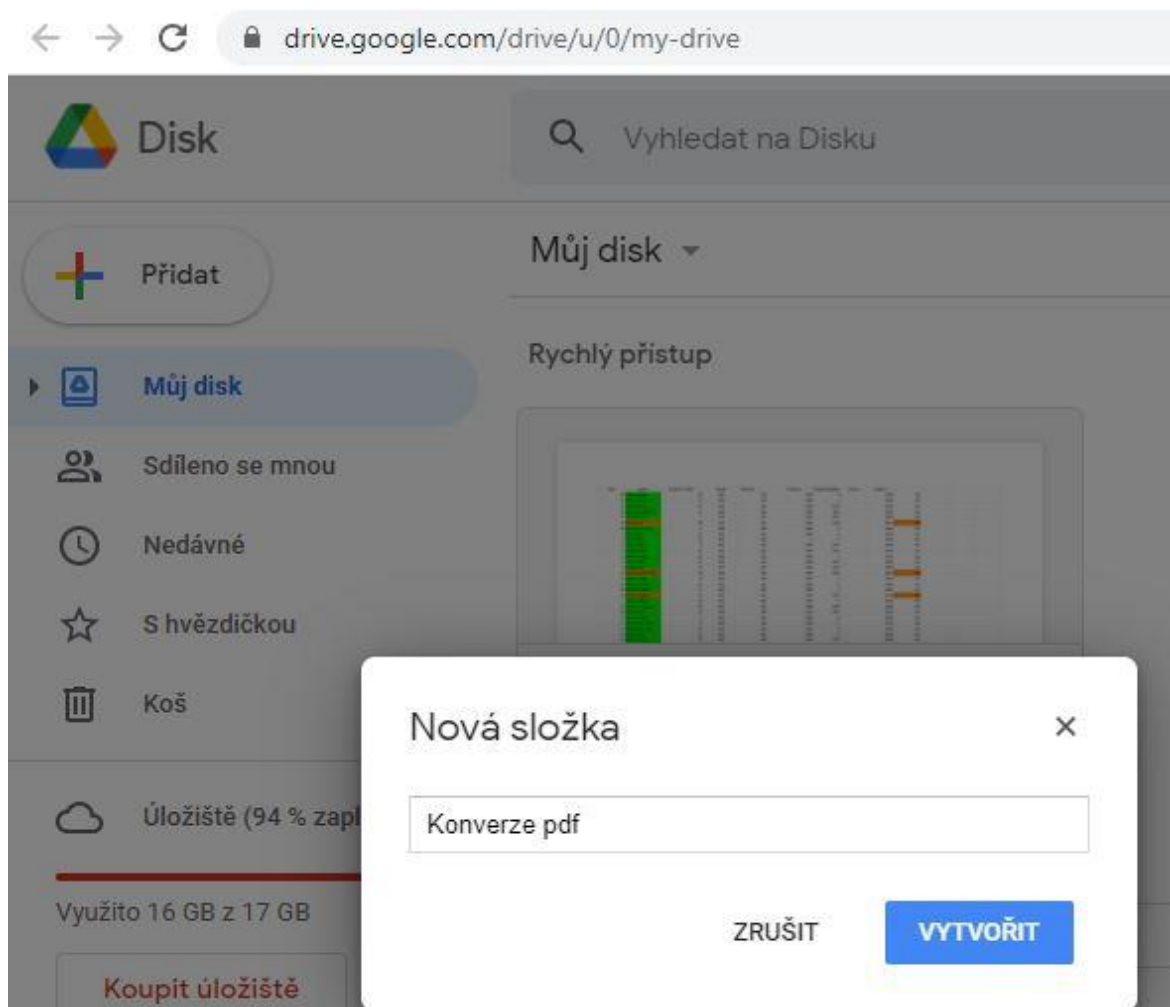
Při nutnosti uložit větší množství souborů ve Wordu, Excelu, případně Powerpointu, je velkým problémem tyto dokumenty uchovávat lokálně a vyhledávat v nich. Zároveň je problém, že se tyto soubory mohou na různých zařízeních zobrazovat různě. Jedním z řešení tohoto problému je služba Google Docs, která umožňuje dokumenty upravit a následně exportovat do pdf formátu.

Jednou z možností konverze dokumentů do formátu pdf je využití služby Google Docs. Tento převod je umožněn pomocí tiskové služby operačního systému Windows nebo služby Bonjour operačního systému MacOS. Google Docs také nabízí možnost přímého exportu souboru do formátu pdf. Tato funkce přímého exportu umožňuje hromadné převedení souborů do formátu pdf, a to včetně souborů ve formátech Open Office nebo formátu HTML.

4.2.1.1 Konverze dokumentů pomocí Google Drive

Prvním krokem je vytvoření nové „vstupní“ složky v aplikaci Google Drive, kam je možné před konverzí nahrát veškeré dokumenty, viz Obrázek 2.

Obrázek 2 – Tvorba vstupní složky



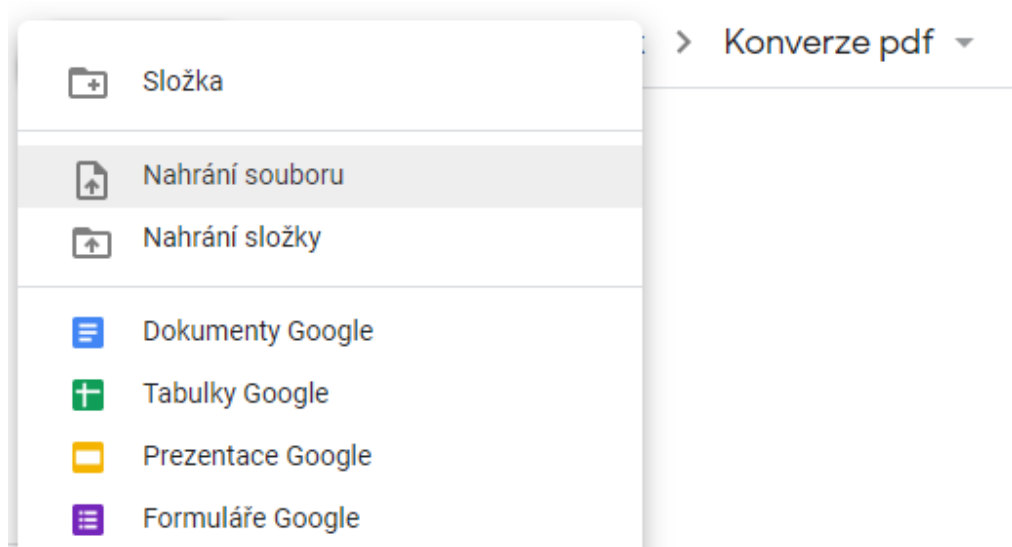
Zdroj: Vlastní zpracování [online].

Podporovány jsou tyto vstupní formáty:

- Microsoft PowerPoint (.ppt, .pps)
- Microsoft Word (.doc, .docx), OpenDocument (.odt) and StarOffice (.sxw).
- Microsoft Excel (csv, .xls, .xlsx) files and OpenDocument Spreadsheet (.ods).

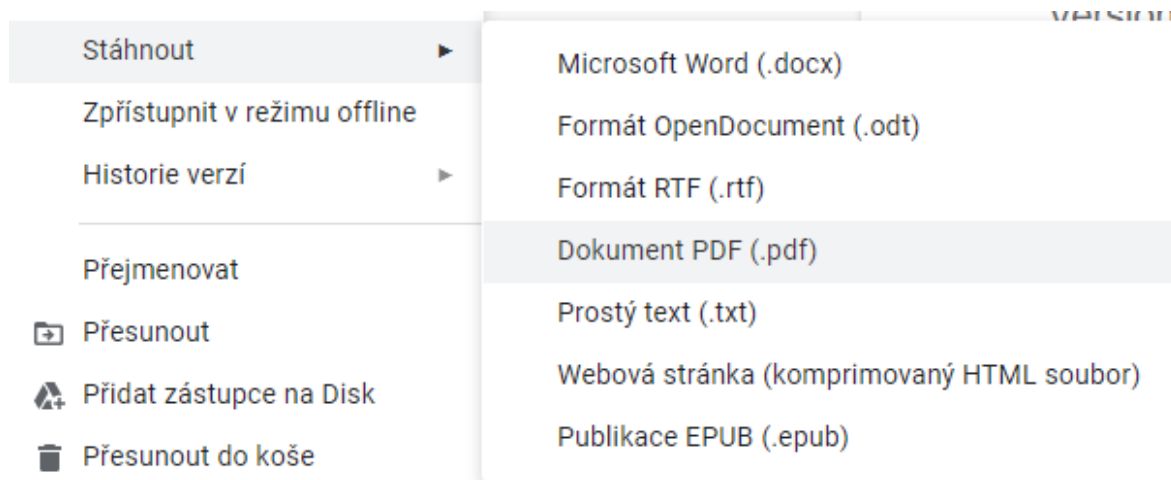
Po nahrání souborů do vytvořené složky v Google Drive, viz Obrázek 3, je potřeba otevřít výchozí nabídku, vybrat „vstupní“ složku a následně přímo Export, kde je na výběr z několika možností výstupního formátu, včetně formátu pdf, viz Obrázek 4. Poté jsou soubory okamžitě převedeny do formátu pdf.

Obrázek 3 – Nahrání vstupních souborů



Zdroj: Vlastní zpracování [online].

Obrázek 4 – Výběr výstupního formátu



Zdroj: Vlastní zpracování [online].

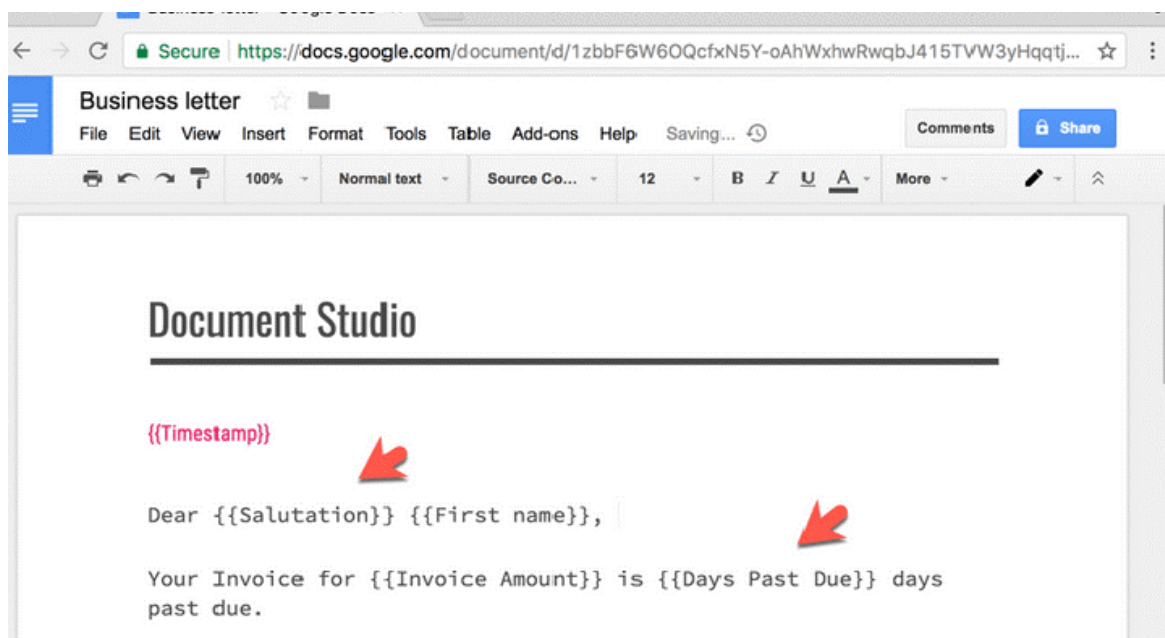
Google Drive nabízí zaslání notifikace e-mailem po dokončení konverze. Tato funkce je vhodná zejména při exportu většího množství dokumentů. Tuto funkci lze spojit se zasláním zkonvertovaných souborů ve formě odkazu na zkomprimovaný soubor.

Tuto funkci konverze nabízí také Google Docs doplněk Google Data Studio, které umožňuje import dat z Excelu nebo CSV formátu a následnou manipulaci s nimi pomocí tagů, kterou lze ovlivnit výsledný vzhled pdf souboru.

4.2.1.2 Konverze dokumentů za použití tagů v Google Docs – tvorba masky

Pro potřeby konverze je nutné vytvořit šablonu (masku), podle které se budou importované soubory konvertovat. Může jít o klasický dokument nebo dokument s řádky a sloupci „Excel“ či prezentaci. Masku je možné formátovat a tím nastavit pozici elementů, např. logo, hlavičku a patičku, podobně jako ve Wordu nebo v Google dokumentu. Do masky je možné zahrnout několik polí předpřipravených pro vyplnění uživatelem, např. pole pro vyplnění jména a příjmení nebo data po splatnosti. Pokud je dokument vytvořen přímo v aplikaci Google Docs, je možné přímo v textu použít funkce, např. Invoice Amount, která vypočítá celkovou sumu ze zadaného sloupce, nebo funkce =UPPER(“{{First Name}}”), která vypíše jméno velkými písmeny ve výsledném dokumentu, viz Obrázek 5.

Obrázek 5 – Přidání tagů pro další práci s proměnnými



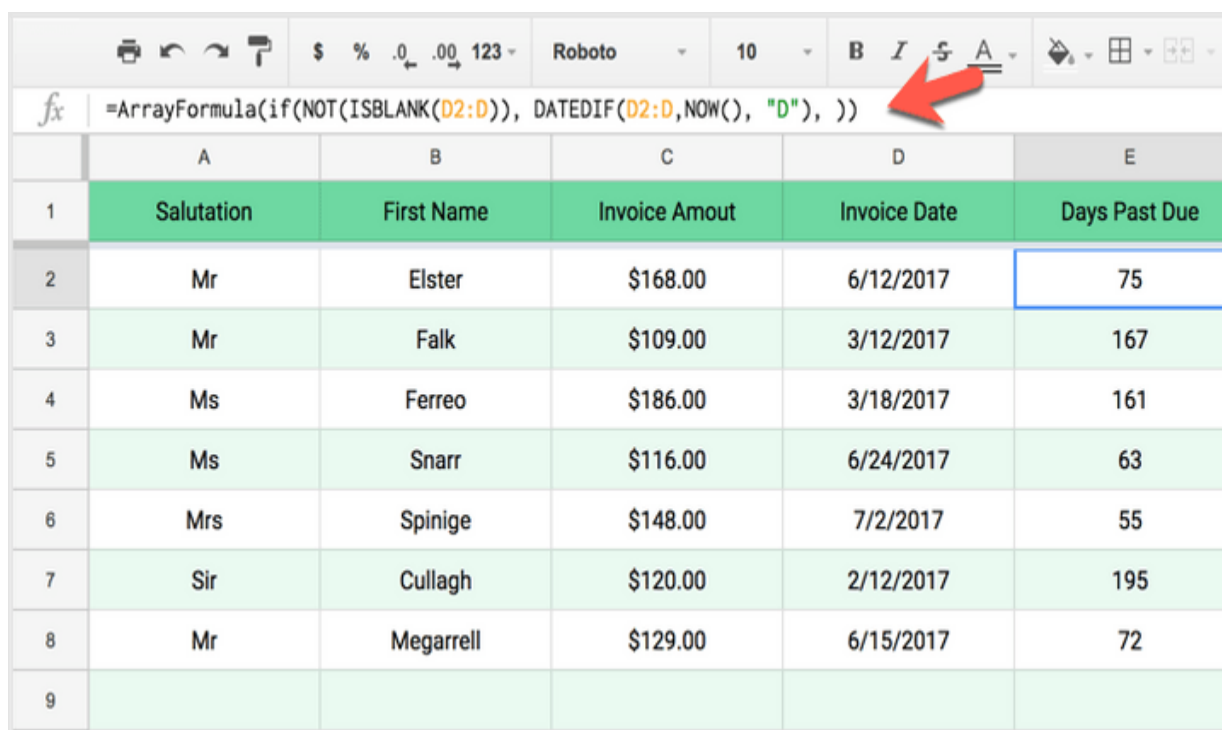
Zdroj: Use Google Docs as a Batch PDF Converter [online]. [cit. 2021-25-02].

<https://www.labnol.org/internet/batch-pdf-converter/10796/>

4.2.1.3 Import dat

Služba Google Docs nabízí více možností importu dat. Data lze ručně vepsat do Google Sheets či je zkopírovat z Excelu, nebo provést přímý import Excel dokumentu nebo CSV dat. Importovaná data se zobrazují jako řádky a sloupce, při importu z Excelu je důležité dodržet názvy sloupců dle původní masky, aby se s danými daty dalo správně dále pracovat a nedocházelo k tvorbě cizích klíčů místo zachování jednoho primárního klíče, viz Obrázek 6.

Obrázek 6 – Párování tagů a dat



The screenshot shows the Google Sheets interface. The formula bar contains the following array formula: `=ArrayFormula(if(NOT(ISBLANK(D2:D)), DATEDIF(D2:D, NOW(), "D"),))`. A red arrow points to the formula bar. Below the formula bar is a table with 5 columns: A (Salutation), B (First Name), C (Invoice Amout), D (Invoice Date), and E (Days Past Due). The table contains 9 rows of data, with the first row being the header and the subsequent 8 rows being data rows. The 'Days Past Due' column is highlighted in blue.

	A	B	C	D	E
1	Salutation	First Name	Invoice Amout	Invoice Date	Days Past Due
2	Mr	Elster	\$168.00	6/12/2017	75
3	Mr	Falk	\$109.00	3/12/2017	167
4	Ms	Ferreo	\$186.00	3/18/2017	161
5	Ms	Snarr	\$116.00	6/24/2017	63
6	Mrs	Spinige	\$148.00	7/2/2017	55
7	Sir	Cullagh	\$120.00	2/12/2017	195
8	Mr	Megarrell	\$129.00	6/15/2017	72
9					

Zdroj: Use Google Docs as a Batch PDF Converter [online]. [cit. 2021-25-02].

<https://www.labnol.org/internet/batch-pdf-converter/10796/>

Do dokumentu je možné přidat i vlastní výpočty, např. počet dní po splatnosti, což lze provést odečtením dnešního data od data splatnosti, a tyto údaje pak vytisknout. Rovněž je možné načíst do zvláštního pole data z vlastního NTP serveru a změnit tak výchozí časové pásmo Google účtu.

4.2.1.4 Nastavení aplikace Document Data Studio

S připravenou maskou a vstupními daty je třeba vybrat doplněk Google Document Data Studio. V sekci spojení je nutné zvolit masku, podle které budou data formátována. Tvorba masky je vysvětlena v kapitole 4.2.1.2. Následně je třeba vybrat soubor vstupních dat, která budou formátována, a název výstupního souboru. V tomto názvu souboru je možné použít data ze souboru vybraná pomocí tagů, např. přidání dnešního data. Výchozí exportní formát je formát pdf, je však možné exportovat soubory i do formátu doc, docx nebo odt.

4.2.1.5 Možnost zaslání výstupního souboru do e-mailu

Pomocí doplňku Mail Merge with Attachment je možné nastavit v aplikaci výstup formou zaslání výstupního souboru do e-mailu ihned po vygenerování. Vzhled tohoto e-mailu je možné ovlivnit pomocí vestavěného WYSIWYG editoru, viz Obrázek 7, dalšího doplňku HTML Mail editor nebo použitím validního HTML kódu.

Obrázek 7 – Tvorba šablony e-mailu

Zdroj: Use Google Docs as a Batch PDF Converter [online]. [cit. 2021-25-02].

<https://www.labnol.org/internet/batch-pdf-converter/10796/>

V doplňku Mail Merge with Attachment je možné použít tagy a proměnné, např. pokud uživatel vyplní e-mail, je exportovaný dokument na tento e-mail automaticky zaslán. Tento doplněk nabízí rovněž standardní funkce e-mailu jako přidání dalších příjemců pomocí polí CC (kopie) nebo BCC (skrytá kopie). Tělo a předmět zprávy může obsahovat tagy ze souboru, např. číslo faktury nebo kolik dní je po splatnosti.

4.2.1.6 Uložení souborů na Google Drive

Pro uložení výstupního souboru na Google Drive stačí při exportu tento doplněk vybrat včetně způsobu uložení v adresářové struktuře. Pomocí tagů je možné rozřídít soubory a uložit je tak do odpovídajících adresářů, např. podle shodujícího se PSČ nebo data vytvoření objednávky, např. `{{Zip Code}}\{{Date of Order}}`.

4.2.1.7 Spuštění generování dokumentu

Po dokončení nastavení je možné konverzi spustit. Proces konverze je možné sledovat v okně prohlížeče včetně následné informace, zda byl soubor zaslán na e-mail. Služba Google Docs nabízí i funkci pravidelné konverze, tedy generování dokumentů například každou hodinu, což je vhodné, pokud je dokument průběžně aktualizován nebo upravován několika uživateli najednou. Tuto funkci lze rovněž využít pro zálohování dat v dokumentu bez nutnosti manuálního spuštění konverze.

4.2.2 Řešení JotForm

Služba JotForm nabízí vícero možností použití, od podepisování smluv online, vystavování certifikátů na základě uživatelem vyplněných dat až po zaslání několika pdf souborů současně na základě vyplněného online dotazníku.

4.2.2.1 Tvorba JotForm formuláře

Vhodnou šablonu vytvořenou např. pomocí Google Docs nebo vestavěného editoru, viz Obrázek 8, je potřeba propojit s online službou pro export dokumentů, jakou je např. aplikace AutoCrat, viz Obrázek 9 a 10.

Obrázek 8 – Vytvoření JotForm šablony a přidání možnosti exportu

Formulář
Martin Hlaváček

Jméno
Jméno Příjmení

Email
email@email.cz

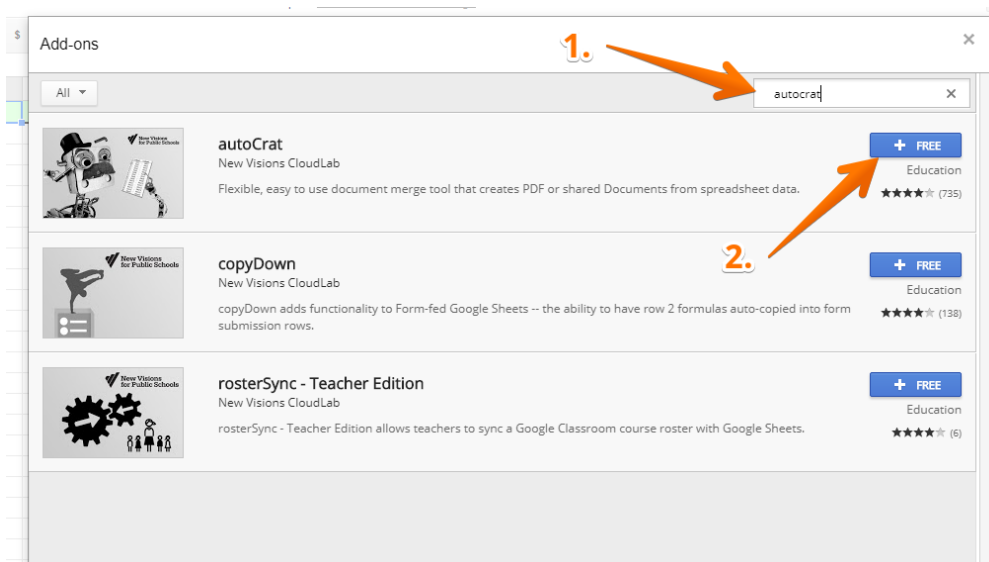
Adresa
Ulice
Číslo popisné
Město Kraj
PSČ

Telefon
Předvolba Telefon

Nahrání souborů
Nahrát

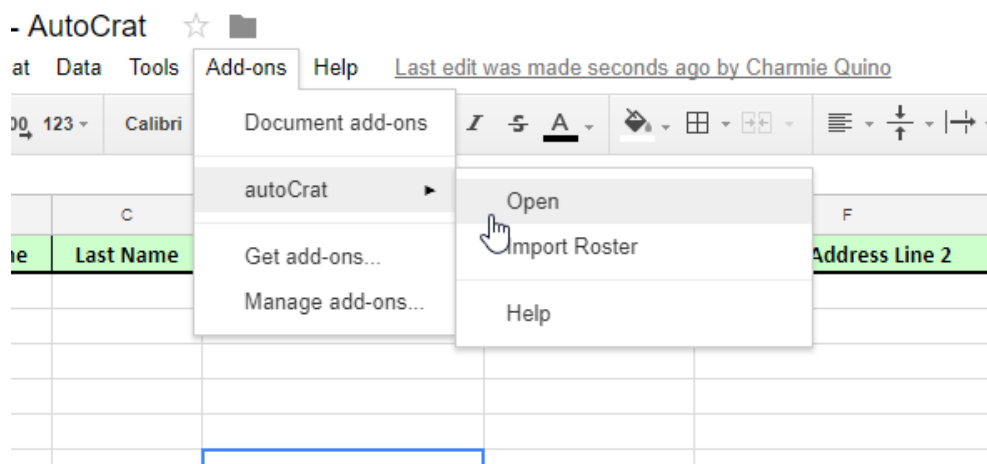
Zdroj: Vlastní zpracování [online]. [cit. 2021-25-02]. <https://form.jotform.com/210611928906355>

Obrázek 9 – Přidání doplňku pro export pdf dokumentu



Zdroj: How to Create a Customized PDF or Google Doc for Your Submissions [online]. [cit. 2021-25-02]. <https://www.jotform.com/help/457-how-to-create-a-customized-pdf-or-google-doc-for-your-submissions>

Obrázek 10 – Aktivace doplňku pro export



Zdroj: How to Create a Customized PDF or Google Doc for Your Submissions [online]. [cit. 2021-25-02]. <https://www.jotform.com/help/457-how-to-create-a-customized-pdf-or-google-doc-for-your-submissions>

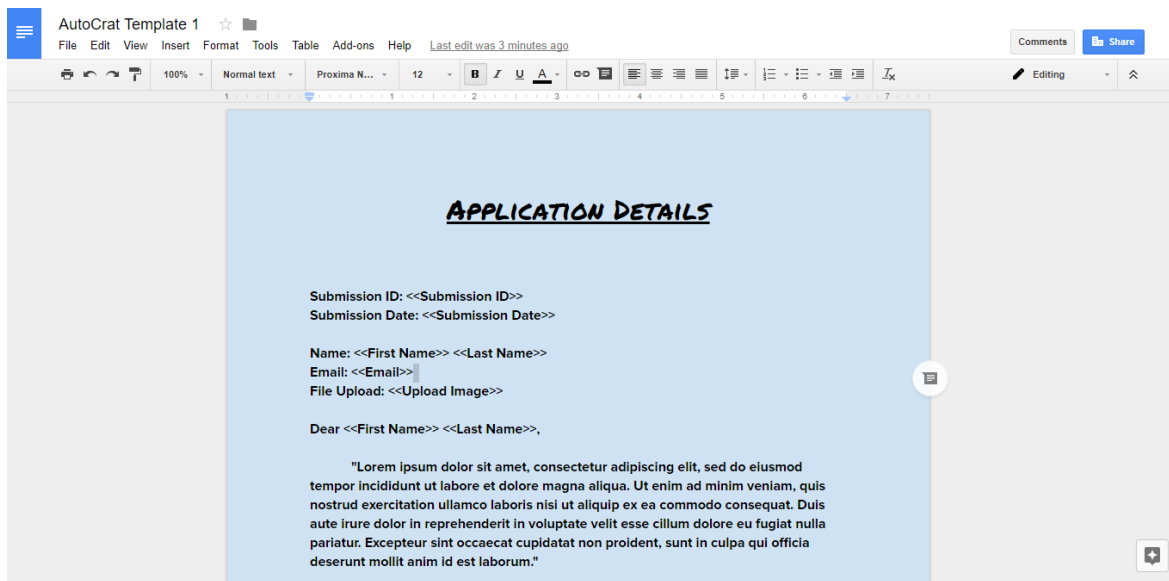
Pro správné fungování služby JotForm je potřeba v jejím prostředí vytvořit tři adresáře, a to hlavní adresář pro nahrávání souborů, adresář pro exportované pdf dokumenty a adresář, který slouží pro uložení šablon.

K již vytvořené šabloně je nutné pomocí tagů, které se vybírají v úvodní šabloně a označují se <<>>, přiřadit potřebná data. To je možné pomocí Google Docs nebo vestavěného editoru. Detailním nastavením této šablony je možné ovlivnit vzhled výstupního dokumentu, např. zvolením barvy pozadí či textu nebo typu písma. Veškerá nastavení se projeví ve výstupním pdf dokumentu, viz Obrázek 11.

Jednotlivé tagy je potřeba předat doplňku AutoCrat a přiřadit tak jednotlivé sloupce v Excel dokumentu k tagům v propojené šabloně, viz Obrázek 12. Jako příklad těchto tagů lze uvést <<Upload Image>>, <<IP>>, <<Submission ID>> sloužící jako primární klíč nebo <<Edit Link>> nesoucí informaci o zdrojovém odkazu uživatele.

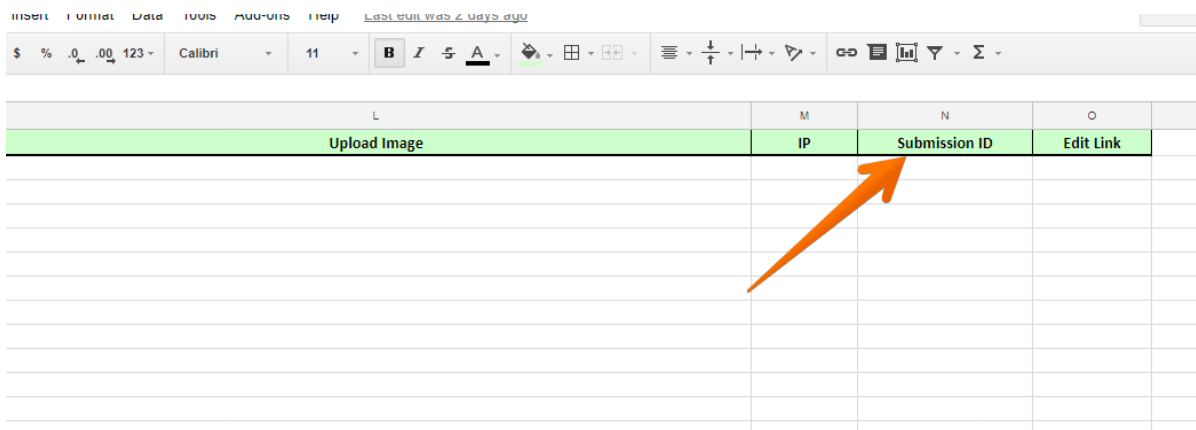
Pro správné fungování aplikace je nutné sloupce z Excel dokumentu provázat s tagy v doplňku AutoCrat, viz Obrázek 13.

Obrázek 11 – Úprava šablony s vloženými tagy



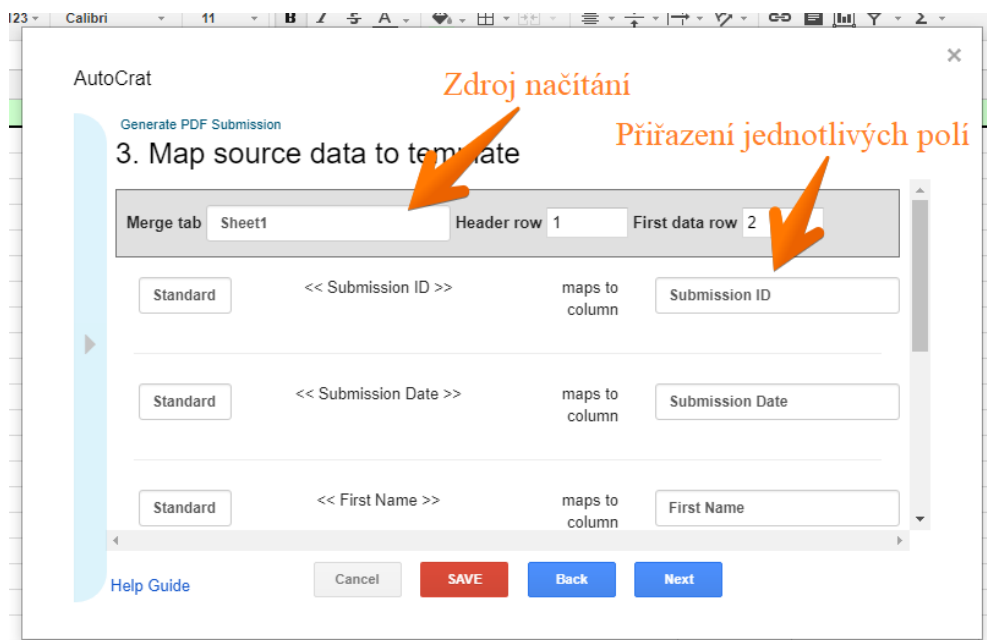
Zdroj: How to Create a Customized PDF or Google Doc for Your Submissions [online]. [cit. 2021-25-02]. <https://www.jotform.com/help/457-how-to-create-a-customized-pdf-or-google-doc-for-your-submissions>

Obrázek 12 - Přiřazení dat ze šablony k tagům



Zdroj: How to Create a Customized PDF or Google Doc for Your Submissions [online]. [cit. 2021-25-02]. <https://www.jotform.com/help/457-how-to-create-a-customized-pdf-or-google-doc-for-your-submissions>

Obrázek 13 - Přiřazení tagů ze šablony k tagům v datech doplňku

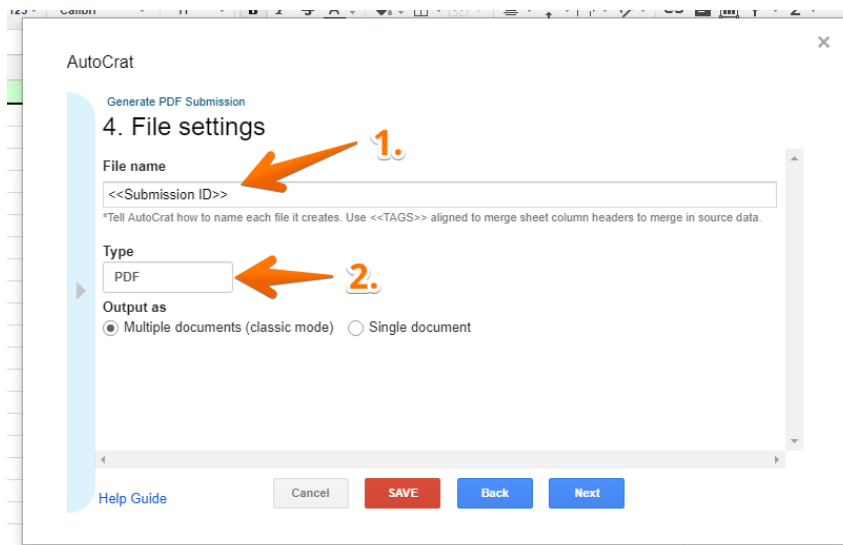


Zdroj: Vlastní zpracování [online].

Pro pojmenování výstupního souboru lze použít tagy ze šablony, např. číslo dokumentu. Dále je možné vybrat, do kterého formátu bude daný soubor exportován, viz Obrázek 14.

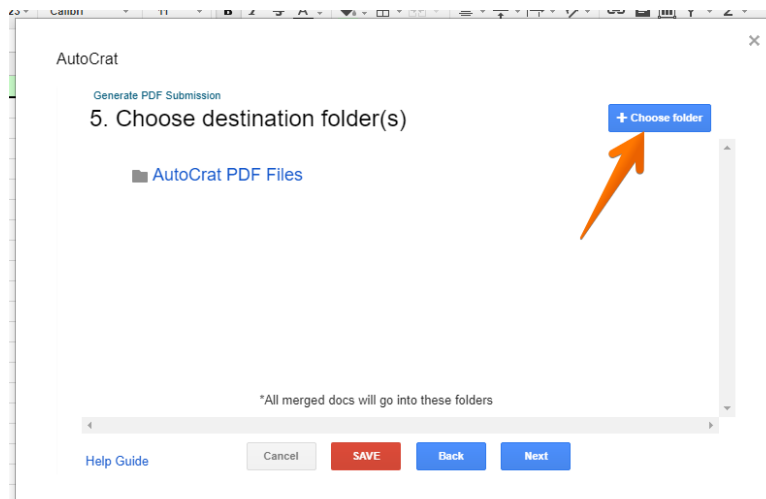
Následně je třeba exportovaný soubor uložit do vlastního adresáře na Google Drive, viz Obrázek 15, aby jej bylo možné odeslat e-mailem.

Obrázek 14 – Pojmenování výstupního souboru



Zdroj: How to Create a Customized PDF or Google Doc for Your Submissions [online]. [cit. 2021-25-02]. <https://www.jotform.com/help/457-how-to-create-a-customized-pdf-or-google-doc-for-your-submissions>

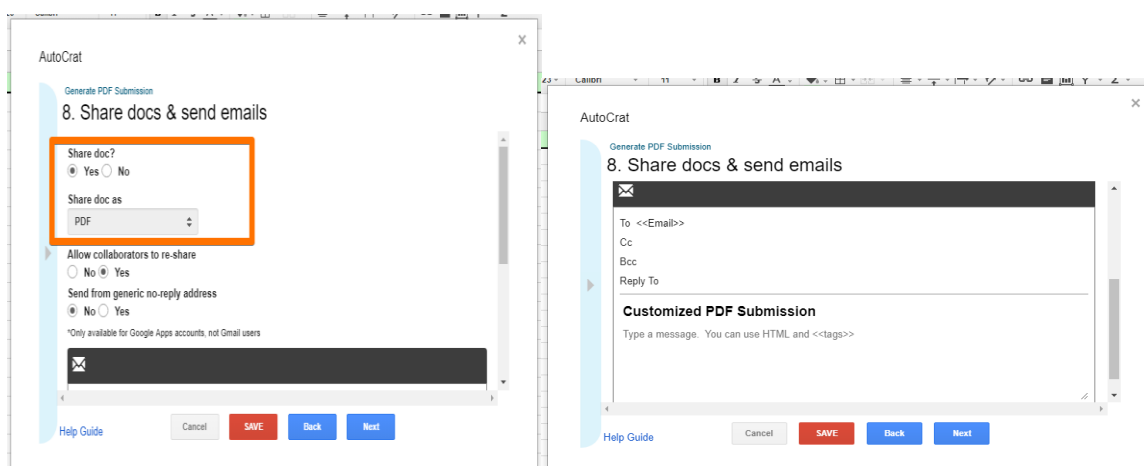
Obrázek 15 – Uložení exportovaného souboru do adresáře



Zdroj: How to Create a Customized PDF or Google Doc for Your Submissions [online]. [cit. 2021-25-02]. <https://www.jotform.com/help/457-how-to-create-a-customized-pdf-or-google-doc-for-your-submissions>

Doplněk AutoCrat nabízí také možnost sloučení více dokumentů do jednoho pdf dokumentu. Výstupní soubor ve formě pdf dokumentu lze odeslat příjemci, jehož e-mailová adresa byla na základě tagu načtena z dokumentu, i více příjemcům současně pomocí polí pro kopii a skrytou kopii, viz Obrázek 16. Služba JotForm umožňuje nastavit plán, jak často celý proces konverze včetně zasílání souborů e-mailem spouštět.

Obrázek 16 – Export a zaslání exportovaného souboru na e-mail



Zdroj: How to Create a Customized PDF or Google Doc for Your Submissions [online]. [cit. 2021-25-02]. <https://www.jotform.com/help/457-how-to-create-a-customized-pdf-or-google-doc-for-your-submissions>

5. Vlastní práce

Tato kapitola se zabývá konkrétním popisem potřebných komponent k vytvoření aplikace pro generování pdf dokumentu dle potřeb podniku. V této kapitole jsou shrnuty požadavky jednotlivých skupin uživatelů (administrátoři, koncoví uživatelé) a je vysvětlena adresářová struktura, konfigurace aplikace a její samotné fungování. Na závěr je představeno rozhraní aplikace.

5.1 Představení podniku

Webový portál isix.cz je na trhu od roku 2016. Jeho specializací je primárně dovoz elektroniky, z níž některé typy produktů nebo konkrétní značky k nám nejsou dováženy jinými subjekty. Tento internetový obchod používá e-shopové řešení od firmy Shoptet, které ale neumožňuje úpravu kompletačního listu. Nemožnost zvětšení a zvýraznění textu v případě objednání více kusů v jedné objednávce způsobovalo značnou chybovost při kompletaci objednávek.

5.2 Představení problému a jeho možného řešení

Při kompletaci objednávek bylo zjištěno, že v podniku isix.cz vzniká značné množství chyb, např. zasláním jednoho kusu zboží místo kusů dvou či zasláním nesprávného typu zboží. Jako hlavní příčina byla vyhodnocena malá velikost písma textu bez dalšího zvýraznění v případě objednání více kusů na vytištěném kompletačním listu objednávky. Při přiřazování štítků s adresou zákazníka k zásilkám docházelo ke zdržení vlivem příliš malého písma textu doručovací adresy zákazníka na kompletačním listu objednávky. Dalším problémem byla absence číslování stránek a chybějící čárový kód pro snímání objednávky čtečkou. Také nebylo možné zajistit seřazení jednotlivých položek na kompletačním listu objednávky, zboží se řadilo náhodně, což znemožňovalo rychlé vychystání zboží z jinak již seřazeného skladu.

Jako vhodné řešení byla zvolena úprava výstupního souboru k tisku kompletačního listu objednávky, která zajistí rychlejší a méně chybovou kompletaci objednávek.

Důvodem k výběru řešení formou úpravy kompletačního listu objednávky namísto zvolení elektronického řešení kompletace objednávek byly především nízké vstupní a udržovací náklady a zachování stávajícího způsobu kompletace objednávek, díky čemuž není nutné zaučovat personál. Výhodou tohoto řešení je možné přenesení kompletace objednávek na jiné pracoviště, například při kompletaci objednávek na veletrzích nebo při osobním dodání.

5.3 Trend momentálního vývoje

Momentální trend směřuje k vývoji online cloudových aplikací, kde jsou data spravována třetí stranou, například služby od firem Google nebo JotForm. „Díky cloudovému řešení odpadnou náklady na firemní hardware a software, které bývají extrémně vysoké. S přechodem na cloud firma platí pouze provozní náklady v podobě měsíčního poplatku za službu. Navíc s cloudem vaše IT systémy nikdy nezestárnou, protože všechny komponenty pravidelně upgradujeme.“ (VODAFONE, 2019)

Nevýhodou cloudových aplikací je jejich nižší zabezpečení, respektive menší kontrola nad daty podniku a osobními údaji zákazníků, a také závislost na cizím technickém řešení a jeho infrastruktuře včetně jeho dalších bezpečnostních rizik vyplývajících z infrastruktury poskytovatele. Nebezpečím je také záměrný prodej dat jinému subjektu. (RYLIYCH, 2012) Nevýhodou cloudového řešení jsou méně rozsáhlé možnosti úpravy na míru a nevhodné rozmístění některých elementů.

5.4 Představení aplikace

V případě, že je třeba dávkově zpracovat velké množství dat, která není vhodné předávat třetí straně, je nutné jít cestou vývoje vlastní aplikace. Výhodou takové aplikace je téměř neomezená životnost a jistota funkčnosti této aplikace i v případě vyšších nákladů na provoz, chod podniku tak neohrozí náhlé ukončení aplikace, protože se třetí straně již nevyplatí tuto službu provozovat (jako například služba Google+, která skončila ze dne na den). (CNEWS, 2019)

5.4.1 Specifikace požadavků

„Pro základní funkčnost a bezpečnost aplikace je třeba dodržet:

- Přizpůsobitelnost aplikace
- Ochrana aplikace proti neoprávněné manipulaci
- Ověření vstupních dat
- Nutnost přihlášení k administrační část
- Rozlišení uživatele a administrátora
- Funkční databázová část bez tvorby duplicit a zapisování dat kam nepatří
- Statistiky přihlášení a ukládání IP adres přihlášených administrátorů
- Kód i výstup z aplikace musí být validní“ (TRNKA, 2007, s.12)

5.4.2 Vývojář

„Pro vývoj a funkčnost celé aplikace je kritické oddělit PHP část se skripty od uživatelsky přístupné části s (X)HTML kódem. Toto rozdělení zabrání poškození aplikace uživatelem a je ideální pro hledání případných chyb nebo nefunkčnosti aplikace. Dalším přínosem tohoto rozdělení je zpřehlednění systému a možnost snazšího rozšiřování systému bez rušení uživatele. Z tohoto důvodu je aplikace tvořena modulárně, jako online aplikace, kdy je možné kdykoliv aplikaci přenést celou na jiný server. Na rozdíl od offline aplikace je možné ji na jiném serveru testovat s novou knihovnou bez zásahu do původní aplikace, která může mezitím uživateli dále sloužit.“ (TRNKA, 2007, s.24)

5.4.3 Koncový uživatel neboli návštěvník

Koncový uživatel neboli návštěvník nepřichází do styku s vnitřní strukturou kódu a pohlíží na aplikaci jako na celek, přestože používá pouze uživatelské prostředí. Aplikaci využívá pouze k předem navrženému účelu, a to k převodu vybraných dat objednávek v xlsx formátu na tisknutelný pdf formát. Není žádoucí, aby koncový uživatel získal přístup ke kódu uloženém na FTP. Uživatel má přístup pouze k uživatelskému prostředí aplikace přes WWW síť.

5.4.4 Bezpečnost

Hlavním požadavkem je zabezpečení aplikace především proti útokům konkurence, čemuž se lze vyhnout použitím vlastní infrastruktury. Oproti konkurenčním cloudovým řešením nabízí vlastní řešení možnost umístit aplikaci na libovolný server včetně vlastního, který nemusí být veřejně přístupný, což zajišťuje vyšší bezpečnost proti jiným řešením, u kterých kód aplikace není ve vlastnictví podniku. Bezpečnost vlastních aplikací je rovněž vyšší než bezpečnost aplikací s hybridní infrastrukturou, jež využívají cloudovou infrastrukturu pouze z části. (TRNKA, 2007)

5.4.5 Adresářová struktura

Z hlediska zabezpečení je nutné navrhnout správnou adresářovou strukturu s předem zvolenou strukturou pro ukládání dat a skriptů. Výpis adresářové struktury aplikace nahrané na FTP serveru vypadá následovně:

```
/
/pdfconverter/
./Classes/
./create_pdf.php/
./index.php/
./Classes/PHPExcel
./Classes/TCPDF
./Classes/PHPExcel.php
./Classes/config.php
./Classes/tcpdf.php
```

Jednotlivé adresáře obsahují následující části aplikace:

- /pdfconverter je kořenový adresář programu. Obsahuje pouze dva základní PHP skripty, které se volají z prohlížeče: soubory index.php a create_pdf.php, a adresář Classes obsahující další knihovny, které jsou potřebné pro běh aplikace a jejích součástí.
- /pdfconverter/index.php zajišťuje generování uživatelského rozhraní webové aplikace.
- /pdfconverter/create_pdf.php načítá knihovnu TCPDF a její PHP skript config.php, čímž zajišťuje generování výstupního pdf dokumentu.

- /Classes/PHPExcel je adresář, který obsahuje soubory pro správné fungování knihovny PHPExcel zajišťující správné načtení xlsx souboru. (PHPExcel, 2013)
- /Classes/TCPDF je adresář obsahující potřebné komponenty pro TCPDF knihovnu. (TCPDF, 2002)
- /Classes/PHPExcel.php je PHP skript, který zajišťuje konfiguraci a nastavení knihovny PHPExcel. (PHPExcel, 2013)
- /Classes/config.php je skript zajišťující generování obsahu pdf dokumentu pomocí knihovny TCPDF. (TCPDF, 2002)
- /Classes/tcpdf.php je PHP skript, který rozšiřuje nastavení generování pdf dokumentu. (TCPDF, 2002)

5.4.6 Vstupní data a obsluha aplikace

Vstupní data jsou výstupem z krabicového e-shopového řešení Shoptet a nelze je měnit, viz Obrázek 17. Vzhled tisknutelných pdf dokumentů přímo z krabicového Shoptet řešení také není možné měnit, k jeho změně je tedy nutné použít externí aplikaci, která slouží ke konverzi dat.

Obrázek 17 – Vstupní data

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q				
1	code	date	status	nan	currency	exchange	email	phone	billFull	Na	billCompe	billStreet	billHouse	billCity	billZip	billCountr	company	vatid	delivery	Ft	
2	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
3	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
4	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
5	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
6	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
7	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
8	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
9	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
10	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
11	21002378	2021-03-11	Nevyřízen	CZK	1		test@he	+42060312	TEST HEUF NOVÁK, s.	Jablonecká 123 - NEČ	Liberec	46001	Česká rep	2387727	CZ023877	TEST HEUF					
12	21002379	2021-03-11	Nevyřízen	EUR	0.0392156		test@gm	+42198712	Test Test		TEST	11	Piešťany	92101	Slovensko					Test Test	
13	21002379	2021-03-11	Nevyřízen	EUR	0.0392156		test@gm	+42198712	Test Test		TEST	11	Piešťany	92101	Slovensko						Test Test
14	21002379	2021-03-11	Nevyřízen	EUR	0.0392156		test@gm	+42198712	Test Test		TEST	11	Piešťany	92101	Slovensko						Test Test
15	21002380	2021-03-11	Nevyřízen	CZK	1		test@gm	+42060312	Test Test		Test 258	test 258	Vrchlabí	54301	Česká republika						Test Test
16	21002380	2021-03-11	Nevyřízen	CZK	1		test@gm	+42060312	Test Test		Test 258	test 258	Vrchlabí	54301	Česká republika						Test Test
17	21002380	2021-03-11	Nevyřízen	CZK	1		test@gm	+42060312	Test Test		Test 258	test 258	Vrchlabí	54301	Česká republika						Test Test
18	21002381	2021-03-11	Nevyřízen	CZK	1		test@gm	+42060312	test		test 111	101101	Klatovy	339 01	Česká republika						test
19	21002381	2021-03-11	Nevyřízen	CZK	1		test@gm	+42060312	test		test 111	101101	Klatovy	339 01	Česká republika						test
20	21002381	2021-03-11	Nevyřízen	CZK	1		test@gm	+42060312	test		test 111	101101	Klatovy	339 01	Česká republika						test
21	21002382	2021-03-11	Nevyřízen	CZK	1		test@gm	+42060312	Test Test		Test 258	258	Vrchlabí	4 543 02	Česká republika						Test Test

Zdroj: Vlastní zpracování. [cit. 2021-15-03].

Uživatelské rozhraní se skládá pouze z jednoho tlačítka sloužícího k nahrání vybraného Excel souboru s daty ke konverzi, viz Obrázek 18. Po nahrání souboru aplikace automaticky provede konverzi.

Obrázek 18- Frontend aplikace

GENERÁTOR PDF

STAČÍ JEDNOU KLIKNOT!

Vyber Excel (xlsx)

Zdroj: Vlastní zpracování [online]. [cit. 2021-15-03]. <http://www.hlavacek.spicka.cz/pdfconverter/>.

Při konverzi aplikace generuje čárový kód čísla objednávky a přidává ho do kompletačního listu pro následné snazší zpracování objednávky. Na kompletačním listu se zřetelně zobrazuje fakturační a doručovací adresa zákazníka. Jednotlivé položky objednávky jsou řazeny dle kódu zboží. V případě objednání více kusů stejného zboží je tento údaj zobrazen tučně a zvýrazněn podtržením. V patičce je uvedeno číslo stránky kompletačního listu. Následně je kompletační list exportován jako pdf dokument, viz Obrázek 19.


Online verze aplikace je umístěna na adrese:

<http://www.hlavacek.spicka.cz/pdfconverter/>.

5.4.7 Výstup z aplikace

Obrázek 19- Výstup z aplikace

Objednávka č.: **21002378**

Příjemce: NOVÁK, s.r.o. TEST HEUREKA Petr Novák Jablonecká 123 - NEODEŠÍLAT OBJEDNÁVKU 46001 Liberec Česká republika Email: test@heureka.cz Telefon: +420603123456		 21002378 Doručovací adresa: JANA, s.r.o. TEST HEUREKA Jana Svobodová Pražská 123 - NEODEŠÍLAT OBJEDNÁVKU 18001, Jablonec nad Nisou Česká republika		Ze dne: 15.03.2021			
Číslo dodávky	Množství	Cena za kus	Cena	Sleva	DPH %	DPH	Celková cena vč. DPH
P90X3: Trénink pro kondici fitnessové DVD nejnovější Kód: 1	1 ks	825,00 Kč	825,00 Kč	0 %	21 %	173,00 Kč	999,00 Kč
profesionální set 13 dvd insanity všechny partie! Kód: 169	1 ks	1 032,00 Kč	1 032,00 Kč	0 %	21 %	216,00 Kč	1 249,00 Kč
Xiaomi Power Bank 10400 mAh (NDY-02-AD) Silver Kód: 197	5 ks	453,00 Kč	2 268,00 Kč	0 %	21 %	476,00 Kč	2 745,00 Kč
GP200 držák do ruky - boxer Kód: 200	1 ks	165,00 Kč	165,00 Kč	0 %	21 %	34,00 Kč	200,00 Kč
FLIPCASE pro Xiaomi Redmi 4A Global Kód: 21	1 ks	82,00 Kč	82,00 Kč	0 %	21 %	17,00 Kč	100,00 Kč
Sportovní kamera SJCAM™ SJ4000 PLUS CZ MENU Varianta: Barva: Bílá Kód: A	1 ks	3 057,00 Kč	3 057,00 Kč	0 %	21 %	641,00 Kč	3 699,00 Kč
Xiaomi Mi 360° Home Security Camera 2K Pro 5G Wifi Kód: B	1 ks	1 066,00 Kč	1 066,00 Kč	0 %	21 %	223,00 Kč	1 290,00 Kč
AB CARVER PRO Posilovací kolečko Kód: C	1 ks	247,00 Kč	247,00 Kč	0 %	21 %	51,00 Kč	299,00 Kč
PPL, při objednání do 18:00 doručováno další pracovní den.	1	57,00 Kč	57,85 Kč	0 %	21 %	12,15 Kč	70,00 Kč
Online platba Heureka.cz - Běžný bankovní převod	1	0,00 Kč	0,00 Kč	0 %	0 %	0,00 Kč	0,00 Kč
Shrnutí			8 802,48 Kč			1 848,52 Kč	10 651,00 Kč
K ZAPLACENÍ			Kč			Kč	
Poznámka:							

5.4.8 Předpoklady pro spuštění aplikace

Aplikace musí být umístěna v kořenovém adresáři na serveru, který podporuje technologii FTP protokolu a verzi 5.6 jazyka PHP. Aplikace je nezávislá na druhu operačního systému, který uživatel používá.

5.4.9 Popis přiložených souborů

Soubor pdfconverter.zip obsahuje kompletní aplikaci. Pro její instalaci je třeba tento soubor extrahovat a kompletní adresář přenést přes FTP protokol do kořenové složky na PHP server. Vstupní data orders.xlsx obsahují testovací data objednávek z e-shopového řešení Shoptet. Administrátorské úpravy, např. odsazení či umístění textů nebo změnu atributů, je možné provést přímo v okomentovaném kódu v Přílohách 1 až 4 této práce, kde jsou příslušné části kódu označeny `//.. a /***..**//`.

6. Závěr

Zpracování dat je dnes velmi důležitý proces. Pomocí optimalizačních kroků je možné významně snížit chybovost a zvýšit efektivitu kompletace objednávek při téměř stejných nákladech. Vlastní práce se zabývá tvorbou aplikace, která zpracovává data a převádí je na jiný formát dat, který je pro člověka snadno čitelný. Pro analýzu byla využita data webového portálu isix.cz, která se zabývá online prodejem pomocí e-shopu.

Porovnáním různých způsobů konverze lze zjistit, že konkurenční řešení tohoto problému pomocí cloudových aplikací nejsou vhodná, což vedlo ke tvorbě vlastní aplikace pro konverzi dat. Hlavním důvodem nevyužití konkurenčních řešení bylo zabezpečení dat, a to především jejich zabezpečení proti úniku k dalším subjektům. Vlastní aplikace rovněž umožňuje běh lokálně a nabízí tak téměř neomezenou životnost aplikace. Velkou výhodou vlastní aplikace jsou větší možnosti úprav.

Zpracovávání objednávek před expedicí je komplexní činnost a způsob jejich kompletace nikdy nelze považovat za zcela optimální. Vylepšením kompletačního listu se zpracování objednávek výrazně zefektivní. Vzhledem k možnosti kód libovolně upravovat se dále nabízí možnost posílit bezpečnost aplikace, a to například formou zabezpečení veřejné stránky pomocí HTTPS protokolu, autentizace uživatele heslem nebo dvoufázovým ověřením.

7. Seznam použitých zdrojů

CNEWS. 2019. Václavík, Lukáš Facebook ubyla konkurence. Sociální síť Google+ právě skončila. [Online] cnews.cz/, 2. Dubna 2019.

DARIE, C. 2006. AJAX a PHP : tvoříme interaktivní webové aplikace profesionálně. Brno : Zoner Press, 2006. ISBN 80-86815-47-1.

DARPA. 2007. Defense Advanced Research Projects Agency . [Online] 21. Květen 2007. https://web.archive.org/web/20070407064829/http://www.darpa.mil/body/arpa_darpa.html.

DENNIS, A. 2003. Tvorba PDF pomocí Adobe Acrobat : průvodce pro profesionály v DTP a prepressu. Brno : Computer Press, 2003. ISBN 80-7226-718-3..

Dev.virtuemart. 2020. Dev.virtuemart. [Online] 2020. http://dev.virtuemart.net/svn/virtuemart/trunk/virtuemart/components/com_virtuemart/helpers/vmpdf.php

DOBEŠOVÁ, Zdena. 2004. DATABÁZOVÉ SYSTÉMY V GIS. UNIVERZITA PALACKÉHO V OLOMOUCI PŘÍRODOVĚDECKÁ FAKULTA. [Online] 2004. <http://www.dobesova.upol.cz/wp-content/uploads/fulltext/DobesovaDatabaze2014.pdf>.

DOMES, M. 2005. Tvorba internetových stránek pomocí HTML, CSS a JavaScriptu. Kralice : Computer Media, 2005. ISBN 80-86686-39-6..

Dyer, J.R.G. 2008. <https://www.sciencedirect.com/>. sciencedirect. [Online] 2008. http://www.cabdyn.ox.ac.uk/complexity_PDFs/Publications_2009/Consensus%20Decision%20Making%20in%20Human%20Crowds.pdf.

FOROUZAN, Behrouz A. 2009. TCP/IP Protocol Suite. New York : McGraw-Hill, 2009. ISBN 978-0-07-337604-2.

GILMORE, W J. -- POKORNÝ, J. 2011. Velká kniha PHP 5 a MySQL : kompendium znalostí pro začátečníky i profesionály. Brno : Zoner Press, 2011. ISBN 978-80-7413-163-9.

- IETF RFC. 1974.** Vinton Cerf, Yogen Dalal, Carl Sunshine Specification of Internet Transmission Control Program. [Online] Prosinec 1974. <https://tools.ietf.org/html/rfc675>.
- JotForm. 2019.** JotForm. [Online] Prosinec 2019. <https://www.jotform.com/help/457-how-to-create-a-customized-pdf-or-google-doc-for-your-submissions>.
- Labnol. 2009.** Labnol Amit Agarwal. [Online] 26. Říjen 2009. <https://www.labnol.org/internet/batch-pdf-converter/10796/>.
- ORACLE. 2017.** Oracle.com. [Online] 2017. <https://www.oracle.com/cz/database/what-is-a-relational-database/>.
- Ourcodeworld. 2017.** Ourcodeworld Carlos Delgado. [Online] Říjen 2017. <https://ourcodeworld.co/articulos/leer/601/como-implementar-y-usar-fuentes-personalizadas-en-tcpdf>.
- Peterka, Jiří. 1992.** Referenční model ISO/OSI - sedm vrstev. earchiv.cz. [Online] 1992. <https://www.earchiv.cz/a92/a213c110.php3>.
- Php.net. 2020.** php.net. php.net. [Online] 2020. <https://www.php.net/supported-versions.php>.
- PHPExcel. 2013.** PHPExcel. [Online] 2013. <https://archive.codeplex.com/?p=phpexcel>
- POKORNÝ, M. 2005.** PHP nejen pro začátečníky. Kraslice : Computer Media, 2005. ISBN 80-86686-38-8..
- RYLICH, Jan. 2012.** ikaros.cz. Cloudové služby: data i počítače v oblacích. [Online] 2012. <https://ikaros.cz/cloudove-sluzby-data-i-pocitace-v-oblacich>.
- Nurdletech. 2014.**Securing FTP using SSH. [Online] Květen 2014.
- Shoptet.cz. 2021.** Shoptet.cz. [Online] Shoptet.cz, 2021. <https://www.shoptet.cz/>.
- TCPDF. 2002.** TCPDF. TCPDF. [Online] 2002. <https://tcpdf.org/>.
- Trnka, Filip. 2007.** Theses. Redakční systém realizovaný pomocí PHP a MySQL. [Online] 2007. https://theses.cz/id/g794q1/downloadPraceContent_adipIdno_6031.

Tutorialspoint. 2015. [Online] Květen 2015.

https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm.

Vodafone. 2019. vodafone.cz. 10 důvodů, proč firmy přechází na cloud. [Online] 12.

Prosinec 2019. [https://www.vodafone.cz/business-blog/cloud/10-duvodu-proc-firmy-prechazi-na-](https://www.vodafone.cz/business-blog/cloud/10-duvodu-proc-firmy-prechazi-na-cloud/#:~:text=D%C3%ADky%20cloudov%C3%A9mu%20%C5%99e%C5%A1en%C3%AD%20v%C3%A1m%20odpadnou,proto%C5%BEe%20v%C5%A1echny%20komponenty%20pravideln%C4%9B%20upgradujeme)

[cloud/#:~:text=D%C3%ADky%20cloudov%C3%A9mu%20%C5%99e%C5%A1en%C3%AD%20v%C3%A1m%20odpadnou,proto%C5%BEe%20v%C5%A1echny%20komponenty%20pravideln%C4%9B%20upgradujeme](https://www.vodafone.cz/business-blog/cloud/10-duvodu-proc-firmy-prechazi-na-cloud/#:~:text=D%C3%ADky%20cloudov%C3%A9mu%20%C5%99e%C5%A1en%C3%AD%20v%C3%A1m%20odpadnou,proto%C5%BEe%20v%C5%A1echny%20komponenty%20pravideln%C4%9B%20upgradujeme).

W3Techs. 2021 Usage of HTTP/2 for websites. [Online]

<https://w3techs.com/technologies/details/ce-http2>.

W3Techs. 2021. Historical trends in the usage statistics of server-side programming languages for websites. [Online]

https://w3techs.com/technologies/history_overview/programming_language.

Web4u.cz. 2013. [Online] 1. Leden 2013.

<https://helpdesk.web4u.cz/index.php?/Knowledgebase/Article/View/240/54/co-je-to-webhosting>.

8. Přílohy

8.1 Příloha 1 - Struktura kódu, část /index.php

```
<?php
session_start();

require_once dirname(__FILE__) . '/Classes/PHPExcel/IOFactory.php';
date_default_timezone_set('Europe/London');

// V případě chyby se pokusí chybu uživateli vypsát (například chybějící
// knihovnu) a ukončí se.
if (isset($_FILES['file']) AND $_FILES['file']['error'] > 0)
{
    echo "Error: " . $_FILES['file']['error'] . "<br>";
    die();
}

// Načtení xlsx souboru pomocí knihovny PHPExcel.
if (isset($_FILES['file']))
{
    $objPHPExcel = PHPExcel_IOFactory::load($_FILES['file']['tmp_name']);

    $objPHPExcel->setActiveSheetIndex(0);

    $rows = $objPHPExcel->getActiveSheet()-
>toArray(null, true, false, false);

    $_SESSION['data'] = $rows;

    header('Location: create_pdf.php');
}
?>
<!DOCTYPE html>
<html class="h-100">
    <head>
        <meta charset="UTF-8">
        <!-- Tvorba názvu v záložce (například pro oblíbené). -->
        <title>Generátor PDF</title>
        <!-- Načítání uživatelského prostředí (frontendu) vytvořeného pomocí
        https://getbootstrap.com/. -->
        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstr
        ap/4.5.1/css/bootstrap.min.css" integrity=" sha384-
        VCmXjywReHh4PwowAiwNagnWcLh1EJLA5buUprzK8rxFgeH0kww/aWY76TfkUoSX" crossorigin="anonymous">
        <!-- Načtení knihovny pro responzivní vzhled a JavaScriptové knihovny. -->
```

```

    <script src="https://code.jquery.com/jquery-
3.6.0.slim.min.js" integrity=" sha256-
u7e5khyithlIdTpu22PHhENmPcRdFiHRjhAuHcs05RI=" crossorigin="anonymous"></scri
pt>
</head>
<body class="h-100">
  <div class="row h-100">
    <div class="col-sm-12 my-auto">
      <div class="w-25 mx-auto">
        <h2 class="text-center text-
uppercase">GENERÁTOR PDF</h2>
        <h6 class="text-center text-
uppercase">STAČÍ JEDNOU KLIKNOU!</h6>
        <br>
        <form enctype="multipart/form-
data" action="index.php" method="POST" style="margin:0 auto; text-
align:center;">
          <label class="btn btn-success btn-lg" for="file">
<!-- Akceptuje pouze xls a xlsx formáty souborů. -->
          <input id="file" type="file" name="file" style="
display:none;" accept=".xls,.xlsx">
          Vyber Excel (xlsx)
          </label>
        </form>
      </div>
    </div>
  </div>
<!-- Tvorba notifikace pro uživatele a předání souboru form. -->
  <script type="text/javascript">
    $('input[name=file]').change(function() {
      $(this).closest('form').submit();
      $(this).parent().html('Nahrávám, prosím vyčkejte');
    });
  </script>
</body>
</html>

```

8.2 Příloha 2 - Struktura kódu, část /create_pdf.php

```
<?php
session_start() ;
// Načtení potřebných tříd.
require_once('Classes/tcpdf.php');
require_once('Classes/config.php');

// -----
// Tvorba proměnných pro načtení dat, proměnná xlsData drží více dat.
// -----
$codeBill = 0;
$i = 0;
$j = 0;
$xlsData = [];

// Používání stejných informací napříč daným souborem.
// Přidání (postincrement) proměnné i.
// Přidání proměnné i jako číslo stránky do patičky.
// Rozdělování položek do objednávek dle čísel objednávek.

foreach ($_SESSION['data'] as $key => $value)
{
    $xlsData = $value;

    if ($codeBill != $xlsData[0])
    {
        $i++;
        $data[$i]['header'] = $xlsData;
        $codeBill = $xlsData[0];
    }
    if ($xlsData[47] == '0' AND $xlsData[45] == '' AND $xlsData[46] == '')
    {
        $data[$i]['footer'][] = $xlsData;
    }
    else
    {
        $sortingKey = isset($xlsData[43]) ? 'ZZZZZZZZZ' . $j++ : $xlsData[
43];
        $data[$i]['detail'][$sortingKey] = $xlsData;
    }
}

// Načítání xlsx souboru do xlsData a přiřazení jednotlivých dat dle
// hlavičky sloupce, v případě změny pořadí sloupce ze strany Shoptet
// je třeba zde párování sloupců změnit.
```



```

foreach ($data as $key => $value)
{
    $xlsData = $value['header'];

    $code = $xlsData[0];
    $date = date('d.m.Y', strtotime($xlsData[1]));
    $email = $xlsData[5];
    $phone = $xlsData[6];
    $billFullName = $xlsData[7];
    $billCompany = $xlsData[8];
    $billStreet = $xlsData[9];
    $billHouseNumber = $xlsData[10];
    $billCity = $xlsData[11];
    $billZip = $xlsData[12];
    $billCountryName = $xlsData[13];

    $companyId = $xlsData[14];
    $vatId = $xlsData[15];
    $deliveryFullName = $xlsData[16];
    $deliveryCompany = $xlsData[17];
    $deliveryVatId = $xlsData[18];
    $deliveryStreet = $xlsData[19];
    $deliveryHouseNumber = $xlsData[20];
    $deliveryCity = $xlsData[21];
    $deliveryZip = $xlsData[22];
    $deliveryCountryName = $xlsData[23];

    $remark = $xlsData[25];

    // Ovlivnění zobrazení pole vpravo nahoře na dokumentu s rozdílnou adresou.
    $address = "";
    if($billFullName == $deliveryFullName &&
        $billStreet == $deliveryStreet &&
        $billCity == $deliveryCity &&
        $billZip == $deliveryZip &&
        $billCountryName == $deliveryCountryName)
    {
        $deliveryFullName = "";
        $deliveryCompany = "";
        $deliveryStreet = "";
        $deliveryHouseNumber = "";
        $deliveryCity = "";
        $deliveryZip = "";
        $deliveryCountryName = "";
    }
    else
    {

```

```

        $deliveryZip .= ",";
        $address ="Doručovací adresa:";
    }

// Přidání měny (převod EUR, USD na symboly).
$currency = 'Kč';
    if ($xlsData['3'] == 'EUR')
        $currency = '€';
    if ($xlsData['3'] == 'GBP')
        $currency = '£';
    if ($xlsData['3'] == 'USD' OR $xlsData['3'] == 'AUD')
        $currency = '$';

// Přidání stránky.
$pdf->AddPage();

// Zarovnání textu.
$padding = 6;

// Seřadí detail objednávky (položky v objednávce).
if (isset($value['detail']) > 0 )
    ksort($value['detail'], SORT_STRING);

// -----
// Tvorba obsahu.
// -----
// Tvorba čárového kodu.
$barcodeStyles = array(
    'position' => 'L',
    'align' => 'C',
    'cellfitalign' => '',
    'hpadding' => '2px',
    'fgcolor' => array(0,0,0),
    'text' => true,
    'font' => 'helvetica',
    'fontsize' => 12,
    'stretchtext' => 0
);

$barcode =
    '<tcpdf method="write1DBarcode" params="' .
    $pdf-
>serializeTCPDFtagParameters([$code, 'C128', '', '', 45, 20, 0.4, $barcodeSt
yles, 'N']) .
    '" />';

```

```

// Generování textu s číslem objednávky, byla použita metoda EOD - HEREDOC
// string bez středníku.
$orderId = <<<EOD
    <span style="font-size:15px;">Objednávka č.:</span>
    <span style="font-size:19px; font-weight: bold: ">{$code}</span>
EOD;

// Umístění a načtení fakturační adresy z proměnných.
$billingAddress = <<<EOD
<style>
    h4 { font-weight: normal; font-size: 13px;}
    strong { font-size: 14px;}
    span { font-size: 13px;}
</style>
<table border="0" cellspacing="0" cellpadding="{$padding}">
    <tr>
        <td><h4>Příjemce:</h4>
            <strong>{$billCompany} <br />
                {$billFullName} <br />
                {$billStreet} {$billHouseNumber}<br />
                {$billZip} {$billCity}<br />
                {$billCountryName}<br />
            </strong>
            <br /><span>Email: {$email}<br />
                Telefon: {$phone}
            </span>
        </td>
    </tr>
</table>
EOD;

// Umístění čárového kodu a načtení proměnné date (objednávka ze dne).
$barcodeAndDate = <<<EOD
<style>
    table { border-right: 1px solid black; border-top:1px solid black; }
    td { text-align: right; }
</style>
<table border="0" cellspacing="0" cellpadding="{$padding}">
    <tr>
        <td>{$barcode}</td>
        <td>Ze dne: {$date}</td>
    </tr>
</table>
EOD;

// Zobrazení doručovací adresy pouze pokud je rozdílná od fakturační.
$shippingAddress = <<<EOD

```

```

<style>
  h4 { font-weight: normal; font-size: 13px;}
  table { border-right: 1px solid black; border-bottom: 1px solid black;}
  table td { height: 160px;}
  strong { font-size: 14px;}
  span { font-size: 13px;}
</style>
<table border="0" cellspacing="0" cellpadding="{$padding}">
  <tr>
    <td><h4>{$address}</h4>
      <br /><strong>{$deliveryCompany}<br />
      {$deliveryFullName}<br />
      {$deliveryStreet} {$deliveryHouseNumber}<br />
      {$deliveryZip} {$deliveryCity}<br />
      {$deliveryCountryName}
    </strong>
  </td>
</tr>
</table>
EOD;

// Umístění a zobrazení hlavičky tabulky nad položkami.
$tableHeader = <<<EOD
  <tr class="tableHeader">
    <th width="300" class="first">Číslo dodávky</th>
    <th width="45">Množství </th>
    <th width="45" align="right" >Cena za kus</th>
    <th width="45" align="right" >Cena</th>
    <th width="45" align="right" >Sleva</th>
    <th width="45" align="right" >DPH %</th>
    <th width="45" align="right" >DPH</th>
    <th width="68" align="right" class="last">Celková cena vč. DPH</th>
  </tr>
EOD;

// Přidání položek, pouze pokud objednávka nějaké položky obsahuje,
// včetně podpory desetinných míst float/double.
$tableBody = null;
$subTotal[1] = 0;
$subTotal[2] = 0;
$subTotal[3] = 0;

if (isset($value['detail']) > 0 )
{
  foreach ($value['detail'] as $key => $val)
  {
    $subTotal[1] = number_format(((double)$val[36],2,',',' '));
  }
}

```

```

$subTotal[2] = number_format((double)$val[37],2,',',' ');
$subTotal[3] = number_format((double)$val[39],2,',',' ');

$val[50] = number_format((double)$val[50],2,',',' ');
$val[54] = number_format((double)$val[54],2,',',' ');
$val[55] = number_format((double)$val[55],2,',',' ');
$val[53] = number_format((double)$val[53],2,',',' ');

$style = ((int)$val[42] > 1) ? 'style=" font-weight: bold; text-
decoration: underline; font-size: 12px;"' : null;
// Načtení čísla produktu (umístění a kódu zboží) a počtu kusů.
$variant = (isset($val[44])) ? '<br />&nbsp;&nbsp;&nbsp;Varianta: ' . $va
l[44] : '';

$itemCode = (isset($val[43])) ? '<br />&nbsp;&nbsp;&nbsp;Kód: ' . $val[43
] : '';
// Načtení cen položek a jejich tisk.
$tableBody .= <<<EOD
<tr class="tableBody">
  <td class="first">{$val[41]}{$variant}{$itemCode}</td>
  <td {$style}>{$val[42]} {$val[46]} </td>
  <td align="right" >{$val[50]} {$currency}</td>
  <td align="right" >{$val[54]} {$currency}</td>
  <td align="right" >0 %</td>
  <td align="right" >{$val[52]} %</td>
  <td align="right" >{$val[55]} {$currency}</td>
  <td align="right" class="last">{$val[53]} {$currency}</td>
</tr>
EOD;
  }
}

// Vložení typu platby, např. Dobírka.
$tableShippingAndPayment = null;
if (isset($value['footer']))
{
  foreach ($value['footer'] as $key => $val)
  {
    $val[50] = number_format((double)$val[50],2,',',' ');

    $tableShippingAndPayment .= <<<EOD
    <tr class="tableShippingAndPayment">
      <td align="left" class="first" >{$val[41]}</td>
      <td align="left" >{$val[42]} {$val[46]} </td>
      <td align="right" >{$val[50]} {$currency}</td>
      <td align="right" >{$val[54]} {$currency}</td>
      <td align="right" >0 %</td>

```

```

        <td align="right" >{$val[52]} %</td>
        <td align="right" >{$val[55]} {$currency}</td>
        <td align="right" class="last">{$val[53]} {$currency}</td>
    </tr>
EOD;
    }
}

// Součet cen položek.
$tableFooter = <<<EOD
<tr class="tableFooter">
    <td width="300" align="left" class="first" >Shrnutí <br /> <strong> K ZA
    PLACENÍ</strong></td>
    <td width="45" align="left" > </td>
    <td width="45" align="right" ></td>
    <td width="45" align="right" >{$subTotal[1]} {$currency}</td>
    <td width="45" align="right" ></td>
    <td width="45" align="right" ></td>
    <td width="45" align="right" >{$subTotal[2]} {$currency}</td>
    <td width="68" align="right" class="last"><strong style="font-
size:9pt"> {$subTotal[3]} {$currency}</strong></td>
</tr>
EOD;

// Ohraničení textu.
$padding = 1;
$table = <<<EOD
    <style>
        table {font-size:10px;}
        .tableHeader th { border-bottom: 1px solid black; border-
top: 1px solid black; font-weight: bold; }
        .tableHeader th.first { border-left: 1px solid black; }
        .tableHeader th.last { border-right: 1px solid black; }

        .tableBody td.first{ border-left: 1px solid black; }
        .tableBody td.last{ border-right: 1px solid black; }

        .tableShippingAndPayment td.first{ border-left: 1px solid black; }
        .tableShippingAndPayment td.last{ border-right: 1px solid black; }

        .tableFooter td { border: 1px solid fuchsia; }
        .tableFooter td { border-bottom: 1px solid black; border-
top: 1px solid black; font-weight: bold; }
        .tableFooter td.first { border-left: 1px solid black; }
        .tableFooter td.last { border-right: 1px solid black; }
    </style>

```

```

<table border="0" cellspacing="0" cellpadding="{padding}">
  {tableHeader}
  {tableBody}
  {tableShippingAndPayment}
  {tableFooter}
</table>
EOD;

// Poznámky od zákazníka.
$notes = <<<EOD
<style>
  h4 { font-weight: normal; font-size: 13px;}
  strong { font-size: 14px;}
  span { font-size: 13px;}
</style>
<table border="0" cellspacing="0" cellpadding="{padding}">
  <tr>
    <td><h4>Poznámka:</h4>
      <br /><span>{$remark}</span>
    </td>
  </tr>
</table>
EOD;

// -----
// Metoda z knihovny TCPDF pro tisk buněk a jejich umístění a ohraničení,
// délka ohraničení dle souřadnic, vše uvedeno v milimetrech.
// -----
$pdf->writeHTMLCell(
  null, 8, // šířka a výška
  null, null, // x a y umístění
  $orderId, 0, 1, 0, true, 'R', false);

$pdf->writeHTMLCell(
  90, 70,
  null, null,
  $billingAddress, 1, 0, 0, true, 'J', false);

$pdf->writeHTMLCell(
  92, null,
  $pdf->GetX() - 1, null,
  $barcodeAndDate, 0, 1, 0, true, 'J', true);

$pdf->writeHTMLCell(
  91, 45,

```

```

105, 48,
$shippingAddress, 0, 1, 0, true, 'J', true);

$pdf->writeHTMLCell(
    182, null,
    $pdf->GetX() - 1, null,
    $table, 0, 1, 0, true, 'J', true);

$pdf->writeHTMLCell(
    182, null,
    $pdf->GetX() - 1, null,
    $notes, 0, 1, 0, true, 'J', true);

$x = $pdf->GetX();

$pdf->Line($x, 280, ($x + 180), 280);
$pdf->Line($x, 80, ($x), 280);
$pdf->Line($x+180, 80, ($x + 180), 280);
}

// -----
// Výstupní soubor.
// -----
$pdf->Output('objednavky.pdf', 'I');
?>

```

8.3 Příloha 3 - Struktura kódu, část Classes/config.php

```

<?php
// Rozšíření třídy TCPDF pro lepší umístění hlavičky a patičky,
// vychází z příkladů dokumentace TCPDF.
class MYPDF extends TCPDF
{
    public function Header() {}

    public function Footer() {
        // Umístění textu 15 mm od okraje stránky, hodnoty jsou v mm.
        $this->SetY(-15);
        // Nastavení fontu na Helvetica a výchozí velikosti 12.
        $this->SetFont('helvetica', 'I', 8);
        // Přidání čísla stránky do pdf.
        $this->Cell(0, 10, $this->getAliasNumPage().' / '.$this->
getAliasNbPages(), 0, false, 'C', 0, '', 0, false, 'T', 'M');
    }
}

```



```

// Tvorba nového pdf dokumentu.
$pdf = new MYPDF(PDF_PAGE_ORIENTATION, PDF_UNIT, PDF_PAGE_FORMAT, true, 'UTF
-8', false);

// Umožňuje ovlivnit vlastnosti pdf dokumentu vytvořeného touto aplikací.
$pdf->SetCreator(PDF_CREATOR);
$pdf->SetAuthor('Martin Hlavacek');
// $pdf->SetTitle('TCPDF Example 003');
// $pdf->SetSubject('TCPDF Tutorial');
// $pdf->SetKeywords('TCPDF, PDF, example, test, guide');

// Nastavení výchozích dat hlavičky.
$pdf-
>SetHeaderData(PDF_HEADER_LOGO, PDF_HEADER_LOGO_WIDTH, PDF_HEADER_TITLE, PDF
_HEADER_STRING);

// Nastavení fontu hlavičky a patičky.
$pdf->setHeaderFont(Array(PDF_FONT_NAME_MAIN, '', PDF_FONT_SIZE_MAIN));
$pdf->setFooterFont(Array(PDF_FONT_NAME_DATA, '', PDF_FONT_SIZE_DATA));

// Výchozí font.
$pdf->SetDefaultMonospacedFont(PDF_FONT_MONOSPACED);

// Nastavení odsazení.
$pdf->SetMargins(PDF_MARGIN_LEFT, 15, PDF_MARGIN_RIGHT);
$pdf->SetHeaderMargin(1);
$pdf->SetFooterMargin(PDF_MARGIN_FOOTER);

// Nastavení konců stran.
$pdf->SetAutoPageBreak(TRUE, PDF_MARGIN_BOTTOM);

// Nastavení škálování obrázků.
$pdf->setImageScale(PDF_IMAGE_SCALE_RATIO);

// Nastavení chování stringů, které závisí na jazyku (volitelné).
if (@file_exists(dirname(__FILE__).'/lang/ces.php')) {
    require_once(dirname(__FILE__).'/lang/ces.php');
    $pdf->setLanguageArray($l);
}

// Nastavení výchozího fontu pro podmnožiny pro nenastavené texty.
$pdf->setFontSubsetting(true);

// Nastavení fontu (další možnosti obsažené v knihovně TCPDF).
// $pdf->SetFont('times', '', 9);
// $pdf->SetFont('dejavusans', '', 9);

```

```

// $pdf->SetFont('arial', '', 9);
// $pdf->SetFont('arimo', '', 9);
// $pdf->SetFont('roboto', '', 8);
$pdf->SetFont('freeserif', '', 9);

unset($_SESSION['data'][0]);
?>

```

8.4 Příloha 4 - Struktura kódu, část Classes/tcpdf.php

```

<?php

require_once('TCPDF/tcpdf.php');

// Možnost přidání obrázků.
// define('K_PATH_IMAGES', PATH_HTML . '/images/');
// Nastavení fontu a velikosti písma vychází z příkladů dokumentace TCPDF.

class PDF extends TCPDF {

    private $dataHeader = array();
    private $dataFooter = array();
    private $martinHL = false;

    public function pole($text = "", $zarovnanilev = 15, $style = '', $size = 8, $radkovani = 4, $velikost = 40, $typ = 'C', $align = 'L') {
        $this->setX(0);
        $this->SetFont('helvetica', $style, $size);
        $this->SetLeftMargin($zarovnanilev);
        if ($typ == 'M') {
            $this->MultiCell($velikost, 0, $text, 0, $align);
        } else {
            $this->Cell($velikost, 0, $text, 0, 0, $align);
        }
        if ($radkovani !=0 ){
            $this->Line($radkovani);
        }
        // Možnost nastavení odsazení textu.
        //$this->SetLeftMargin(15);
    }

    public function etiketapopisek($etiketa = "", $pole = "", $zarovnanilev = 15,$styleEtiketa = '',$stylePole = '', $size = 8, $radkovani = 4,$velikost Etiketa = 40,$velikostPole = 40, $typEtiketa = 'C',$typPole = 'C', $alignEtiketa = 'L',$alignPole = 'L') {

```

```

$this->setX(0);
$this->SetFont('helvetica', $styleEtiketa, $size+1);
$this->SetLeftMargin($zarovnanilev);
if($typEtiketa == 'C'){
    $this->Cell($velikostEtiketa, 0, $etiketa, 0, 0, $alignEtiketa);
}else{
    $this->
>MultiCell($velikostEtiketa, 0, $etiketa, 0, $alignEtiketa);
}
$this->SetFont('helvetica', $stylePole, $size+1);

if($typPole == 'C'){
    $this->Cell($velikostPole, 0, $pole, 0, 0, $alignPole);
}else{
    $this->MultiCell($velikostPole, 0, $pole, 0, $alignPole);
}

$this->Line($radkovani);
}

public function group($text = "", $zarovnanilev = 15, $style = '', $size
= 8, $radkovani = 4, $valX = 0, $align = 'L') {
    $this->SetFont('helvetica', $style, $size);
    $this->setX($valX);
    $this->SetLeftMargin($zarovnanilev);
    $this->Write(0, $text, '', 0, $align, true, 0, false, false, 0);
    if ($radkovani !=0 ){
        $this->Line($radkovani);
    }
    $this->SetLeftMargin(15);
}

function setHeaderLogo($image = '', $imageWidth = PDF_HEADER_LOGO_WIDTH)
{
    $this->dataHeader['image'] = $image;
    $this->dataHeader['imageWidth'] = $imageWidth;
}

function setHeaderTitle($title = '', $titleSize = '') {
    $this->dataHeader['title'] = $title;
}

function setHeaderSubTitle($subTitle = '', $subTitleSize = '') {
    $this->dataHeader['subTitle'] = $subTitle;
}

function setHeaderSubTitle2($subTitle = '', $subTitleSize = '') {

```

```

        $this->dataHeader['subTitle2'] = $subTitle;
    }

    function setHeaderTextRight($text) {
        $this->dataHeader['text-right'][] = $text;
    }

    function setFooterTextLeft($text) {
        $this->dataFooter['text-left'][] = $text;
    }

    function setMartinHL($val) {
        $this->martinHL = $val;
    }

    /** Ourcodeworld. 2017. Ourcodeworld Carlos Delgado. [Online]
    Říjen 2017. https://ourcodeworld.co/articulos/Leer/601/como-implementar-y-usar-fuentes-personalizadas-en-tcpdf.*/

    // Nastavení vlastností fontu.
    // function setFontStyle($text){
    // $this->SetFont($this->FontFamily,$text,$this->FontSize);}

    /**
     * Tato metoda je používána k rendrování hlavičky dokumentu,
     * je možné ji zavolat pomocí AddPage().
     * Její chování je možné nastavit ve třídě @public.
     */
    public function Header() {

        $this->SetHeaderData($this->dataHeader['image'], $this-
>dataHeader['imageWidth'], $this->dataHeader['title'], $this-
>dataHeader['subTitle']);
        // Nastavení nové XObject šablony.
        // if ($this->header_xobjid < 0) {
        $this->header_xobjid = $this->startTemplate($this->w, $this-
>tMargin);
        $headerfont = $this->getHeaderFont();
        $headerdata = $this->getHeaderData();
        $this->y = $this->header_margin;
        if ($this->rtl) {
            $this->x = $this->w - $this->original_rMargin;
        } else {
            $this->x = $this->original_lMargin;
        }
    }

```

```

        if (($headerdata['logo']) AND ($headerdata['logo'] != K_BLANK_IMAGE)
) {
            $imgtype = TCPDF_IMAGES::getImageFileType(K_PATH_IMAGES . $headerdata['logo']);
            if (($imgtype == 'eps') OR ($imgtype == 'ai')) {
                $this->
>ImageEps(K_PATH_IMAGES . $headerdata['logo'], '', '', $headerdata['logo_width']);
            } elseif ($imgtype == 'svg') {
                $this->
>ImageSVG(K_PATH_IMAGES . $headerdata['logo'], '', '', $headerdata['logo_width']);
            } else {
                $this->
>Image(K_PATH_IMAGES . $headerdata['logo'], '', '', $headerdata['logo_width']);
            }
            $imgy = $this->getImageRBY();
        } else {
            $imgy = $this->y;
        }

        $cell_height = round(($this->cell_height_ratio * $headerfont[2]) / $this->k, 2);
        // Nastavení počátečního odsazení pro textové buňky.
        if ($this->getRTL()) {
            $header_x = $this->original_rMargin + ($headerdata['logo_width'] * 1.1);
        } else {
            $header_x = $this->original_lMargin + ($headerdata['logo_width'] * 1.1);
        }
        $cw = $this->w - $this->original_lMargin - $this->original_rMargin - ($headerdata['logo_width'] * 1.1);
        // Zarovnání hlavičky doprava.
        $hwr = 50;
        $cw = $cw - $hwr;
        $this->SetTextColorArray($this->header_text_color);
        // Titulek hlavičky.
        $this->SetFont($headerfont[0], 'BI', $headerfont[2] + 1);
        $this->SetX($header_x);
        $this->
>Cell($cw, $cell_height, $headerdata['title'], 0, 1, 'C', 0, '', 0);
        // Přidání textu do hlavičky pomocí stringu.
        $this->SetFont($headerfont[0], 'BI', $headerfont[2]);
        $this->SetX($header_x);

```

```

    $this->
>MultiCell($cw, $cell_height, $headerdata['string'], 0, 'C', 0, 1, '', '', t
rue, 0, false, true, 0, 'T', false);

    $this->SetFont($headerfont[0], 'BI', $headerfont[2]);
    $this->SetX($header_x);
    $this->MultiCell($cw, $cell_height, $this->
>dataHeader['subTitle2'], 0, 'C', 0, 1, '', '', true, 0, false, true, 0, 'T'
, false);

    // Tisk pravé části dokumentu.
    $this->SetFont($headerfont[0], $headerfont[1], $headerfont[2] - 4);
    $this->y = $this->header_margin;
    $this->SetX($header_x + $cw - 2);
    $textRight = "";
    foreach ($this->dataHeader['text-right'] as $key => $value) {
        $textRight .= $value . "\n";
    }
    // Nastavení výšky a šířky řádku a možné parametry.
    // $this->MultiCell($hrWidth, $hrRowHeight, 'Pag. ' .
    // $this->PageNo() . '{nb}', 0, 'R', 0);
    $this->
>MultiCell($hwr, $cell_height, $textRight, 0, 'R', 0, 1, '', '', true, 0, fa
lse, true, 0, 'T', false);

    // Možnost volby odsazení.
    // debug($this->h) + original_rMargin+ footer_margin;
    $this->SetY((2.835 / $this->k) + max($imgy, $this->y));
    if ($this->rtl) {
        $this->SetX($this->original_rMargin);
    } else {
        $this->SetX($this->original_lMargin);
    }
    $this->Cell(($this->w - $this->original_lMargin - $this->
>original_rMargin), 0, '', 'T', 0, 'C');
    $this->endTemplate();
    // }
    $this->Line($this->original_rMargin, 8, ($this->w - $this->
>original_lMargin), 8);
    $this->Line($this->original_rMargin, 26, ($this->w - $this->
>original_lMargin), 26);

    $this->Line($this->original_rMargin, ($this->h - $this->
>footer_margin + 6), ($this->w - $this->original_lMargin), ($this->h -
    $this->footer_margin + 6));

```

```

        $this->Line($this->original_rMargin, 8, 10, ($this->h - $this-
>footer_margin + 6));
        $this->Line(($this->w - $this->original_lMargin), 8, ($this->w -
$this->original_lMargin), ($this->h - $this->footer_margin + 6));

// Tisk hlavičky ze šablony pro oboustranný tisk.
// Dev.virtuemart. 2020. Dev.virtuemart. [Online] 2020.
//http:dev.virtuemart.net/svn/virtuemart/trunk/virtuemart/components/com_vi
//rtuemart/helpers/vmpdf.php
        $x = 0;
        $dx = 0;
        if (!$this->header_xobj_autoreset AND $this->booklet AND (($this-
>page % 2) == 0)) {
// Upravení odstupů pro oboustranný tisk.
            $dx = ($this->original_lMargin - $this->original_rMargin);
        }
        if ($this->rtl) {
            $x = $this->w + $dx;
        } else {
            $x = 0 + $dx;
        }
        // debug($x);
        $this->printTemplate($this-
>header_xobjid, $x, 0, 0, 0, '', '', false);
        if ($this->header_xobj_autoreset) {
// Zresetování hlavičky pro umožnění tvorby dokumentu na každé nové stránce.
            $this->header_xobjid = -1;
        }
    }

    public function Footer() {
        $cur_y = $this->y - 3;
        $this->SetTextColorArray($this->footer_text_color);
        //Nastavení ohraničení buňek
        $line_width = (0.85 / $this->k);

// Možné nastavení ohraničení tisku.
// $this->SetLineStyle(array('width' => 0.85 /
// $this->k, 'cap' => 'butt', 'join' => 'miter',
// 'dash' => 0, 'color' => $headerdata['line_color']));

$this-
>SetLineStyle(array('width' => $line_width, 'cap' => 'butt', 'join' => 'mite
r', 'dash' => 0, 'color' => $this->footer_line_color));
        // Tisk čárového kodu a jeho zarovnání v dokumentu.
        $barcode = $this->getBarcode();
    }

```

```

        if (!empty($barcode)) {
            $this->Line($line_width);
            $barcode_width = round(($this->w - $this->original_lMargin -
$this->original_rMargin) / 3);
            $style = array(
                'position' => $this->rtl ? 'R' : 'L',
                'align' => $this->rtl ? 'R' : 'L',
                'stretch' => false,
                'fitwidth' => true,
                'cellfitalign' => '',
                'border' => false,
                'padding' => 0,
                'fgcolor' => array(0, 0, 0),
                'bgcolor' => false,
                'text' => false
            );
            $this-
>write1DBarcode($barcode, 'C128', '', $cur_y + $line_width, '', (($this-
>footer_margin / 3) - $line_width), 0.3, $style, '');
        }
        if (empty($this->pagegroups)) {
            $pagenumtxt = $this->l['w_page'] . ' ' . $this-
>getAliasNumPage() . ' / ' . $this->getAliasNbPages();
        } else {
            $pagenumtxt = $this->l['w_page'] . ' ' . $this-
>getPageNumGroupAlias() . ' / ' . $this->getPageGroupAlias();
        }
        $this->SetY($cur_y);

        // Tisk textu vlevo.
        $headerfont = $this->getHeaderFont();
        $this->SetFont($headerfont[0], $headerfont[1], $headerfont[2] - 4);
        $this->SetX($this->original_lMargin + 2);
        $textLeft = "";
        foreach ($this->dataFooter['text-left'] as $key => $value) {
            $textLeft .= $value . "\n";
        }

        $cell_height = round(($this-
>cell_height_ratio * $headerfont[2]) / $this->k, 2);
        $hwr = 50;
        // Nastavení výšky a šířky řádku a možné parametry
        // $this->MultiCell($hrWidth, $hrRowHeigt, 'Pag. ' .
        // $this->PageNo() . '{nb}', 0, 'R', 0);
        $this-
>MultiCell($hwr, $cell_height, $textLeft, 0, 'L', 0, 1, '', '', true, 0, fal
se, true, 0, 'T', false);

```



```

$this->SetY($cur_y);

// Tisk čísla stránky do patičky.
if ($this->getRTL()) {
    $this->SetX($this->original_rMargin);
    $this->Cell(0, 0, $pagenumtxt, 'T', 0, 'L');
} else {
    $this->SetX($this->original_lMargin);
    $this->Cell(0, 15, '', 'T', 0, 'R');
}

// Zjištění velikosti stránky dokumentu odečtením levého a pravého
// zarovnání.
$cw = $this->w - $this->original_lMargin - $this->original_rMargin;
// Zarovnání hlavičky doprava.
$hwr = 88;
$cw = $cw - $hwr;

$this->SetY(15);
$this->SetX($cw);
$this->Cell(101, 15, $this-
>getAliasRightShift() ." Pag. ". $pagenumtxt, 0, 0, 'R');

$SITE_TEST = 1;

if ($SITE_TEST != 0) {
    $txt = "T E S T";

    $this->waterMark($txt);
}

if ($this->martinHL) {
    $this->SetAlpha(0.1);
    // Otočení o 45° a vytištění vodoznaku.
    $this->StartTransform();
    // Nastavení rotace.
    // $this->Rotate(0, $myX, $myY);
    $this->SetColor('text', 255, 0, 34);
    $this->SetFont("courier", "B", 60);
    $this->Text(60, 50, "HLAVACEK");
    $this->StopTransform();
    // Resetování průhlednosti na původní hodnotu.
    $this->SetAlpha(1);
}
}

```

```

function waterMark($text) {
    // Zjištění délky a šířky stránky.
    $myPageWidth = $this->getPageWidth();
    $myPageHeight = $this->getPageHeight();
    // Nalezení středu stránky a přizpůsobení textu.
    $myX = ( $myPageWidth / 2 ) - 90;
    $myY = ( $myPageHeight / 2 ) + 75;
    // Nastavení textu tisku na světlou barvu.
    $this->SetAlpha(0.21);
    // Otočení o 45° a tisk vodoznaku.
    $this->StartTransform();
    $this->Rotate(45, $myX, $myY);
    $this->SetColor('text', 255, 0, 34);
    $this->SetFont("courier", "B", 60);
    $this->Text($myX, $myY, $text);
    $this->StopTransform();
    // Resetování průhlednosti na původní hodnotu.
    $this->SetAlpha(1);
}
}

```