

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

System správy stravovacího zařízení

Plzeň 2013

Michal Kacerovský

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 6. května 2013

Michal Kacerovský

Abstract

Catering Facility Management System

This bachelor thesis deals with development of management system for catering facilities. Its main aim is to analyse simultaneously used web application, determine its functional and program weaknesses and on their basis finally choose appropriate technologies and procedures for building new, more flexible and easily extendable system. One of the important aspects is user friendly graphical interface that creates control of the application more comfortable for inexperienced user.

Poděkování

Během vytváření jak teoretické, tak i praktické části této práce mi svými radami, podporou nebo jiným způsobem přispělo několik lidí, kterým bych rád vyjádřil vděčnost. Můj dík patří

vedoucímu práce Ing. Richardu Lipkovi, Ph.D., za cenné rady, věcné připomínky a vstřícný přístup během vytváření této práce,

Haně a Petrovi Markovým, kteří mě seznámili s chodem stravovacího zařízení a poskytli příležitost pro vznik této práce,

Davidu Hrbáčkovi za nekonečné Skype konverzace, které mi zpestřovaly hodiny strávené u počítače,

všem testerům, jmenovitě Martinu Altmanovi, Barboře Dřimalové, Kristýně Hudcové, Kristýně Kacerovské, Olze Kacerovské, Antonínu Kacerovskému, Miroslavu Kubiskovi, Antonínu Steinbergerovi a Magdaleně Žánové,

Mgr. Martinu Jakubíkovi za to, že minimalizoval gramatické a stylistické nedostatky v této práci,

a v neposlední řadě rodině za podporu během studia.

Obsah

1	Úvod	1
2	Analýza původního systému	2
2.1	Programová struktura	2
2.1.1	(X)HTML	2
2.1.2	Kaskádové styly	2
2.1.3	JavaScript a jQuery	2
2.1.4	PHP a mPDF	3
2.1.5	MySQL	3
2.2	Bezpečnost	5
2.2.1	Zabezpečení přenosu dat	5
2.2.2	Ochrana hesel	5
2.2.3	SQL a PHP injection	5
2.3	Funkční celky aplikace	6
2.3.1	Výpisy	6
2.3.2	Příchozí objednávky	6
2.3.3	Řidiči	7
2.3.4	Zákazníci	7
2.3.5	Trasy	7
2.3.6	Objednávky	7
2.3.7	Nastavení	7
2.4	Uživatelské rozhraní	8
2.4.1	Globální aspekty	8
2.4.2	Ovládací prvky	9
3	Návrh nového systému	12
3.1	Obecná charakteristika	12
3.1.1	Rozsah projektu	12
3.1.2	Kontext systému	12
3.1.3	Funkce produktu	12
3.1.4	Autentizace a autorizace	13
3.1.5	Provozní prostředí	13
3.1.6	Omezení návrhu a implementace	14
3.1.7	Systém jako softwarový produkt	15

3.2	Inovace a nové funkce	15
3.2.1	Nástěnka	15
3.2.2	Zákazníci	16
3.2.3	Distribuce	16
3.2.4	Objednávky	17
3.2.5	Finance	18
3.2.6	Stravování	19
3.2.7	Výpisy	19
3.2.8	Nastavení	20
3.3	Koncepce uživatelského rozhraní	22
3.4	Souhrn požadavků na nový systém	24
4	Implementace systému Gadmin	25
4.1	Srovnání a volba technologií	25
4.1.1	Generování HTML obsahu	25
4.1.2	Databáze	26
4.1.3	Dynamika	27
4.1.4	Uživatelské rozhraní	28
4.2	Architektura systému	28
4.2.1	Adresářová struktura Nette, resp. aplikace	28
4.2.2	MVC a MVP vzor	29
4.2.3	Detailní model systému Gadmin	30
4.2.4	Presentery	31
4.2.5	Pohledy	32
4.2.6	Konfigurace frameworku	34
4.2.7	Databáze	35
4.3	Popis obecné funkcionality	36
4.3.1	Navigace a AJAX	36
4.3.2	Autentizace a autorizace	38
4.4	Funkční moduly	39
4.4.1	Zákazníci	39
4.4.2	Distribuce	40
4.4.3	Objednávky	42
4.5	Uživatelské rozhraní	45
4.5.1	Použité techniky a jejich případné nedostatky	45
4.5.2	Nové ovládací prvky	47
4.5.3	Prostředky nápovědy	49
5	Testování	52
5.1	Dostupné metody testování	52
5.2	Testování kompatibility	52
5.3	Manuální testování	53
5.3.1	Výsledky testovacích scénářů	55
5.3.2	Výsledky závěrečného průzkumu	55

5.4	Výsledky a jejich význam pro další vývoj	57
5.4.1	Uživatelské rozhraní, intuitivita	57
5.4.2	Funkcionalita	57
5.4.3	Prostředky nápovědy	58
5.4.4	Další úpravy	58
6	Závěr	60

1 Úvod

V dnešní uspěchané době nemá spousta lidí možnost myslet na pravidelné stravování – buď jim to neumožňuje jejich zaměstnání, docházka do školy anebo jen nechtějí každé odpoledne strávit u sporáku přípravou zítřejšího oběda. Právě pro takové existují společnosti, které zajišťují pravidelný rozvoz pokrmů jak do domácností, tak i do firem.

Zadavatelem této práce je právě jedno z takových zařízení poskytující distribuci obědů a cateringové služby na území okresu Domažlice. Každodenně potřebuje organizovaně evidovat objednávky od skupiny řidičů a výdejen a objednávky přijímané elektronicky (e-mailem či z e-shopu) nebo telefonicky. Ty musí být provázány se seznamem zákazníků a trasami jednotlivých řidičů. Každý zákazník přitom může volit z několika možností balení pokrmů a způsobů jejich úhrady.

Původně vedení stravovacího zařízení využívalo pro účely evidence takových dat aplikaci Excel společnosti Microsoft. S narůstajícím počtem objednávek však nebylo možné nadále udržovat data takového charakteru pouze v souborech tabulkového procesoru – jejich správa byla nesmírně obtížná a časově náročná. Problémy nastávaly především ve chvíli, kdy bylo nutné přidat do tabulek nové zákazníky, staré odebrat, změnit pořadí stanic v trasách řidičů apod.

Pro tyto účely byla v roce 2011 vyvinuta webová aplikace, která měla výše zmíněné úkony administrativním pracovníkům zařízení usnadnit, ušetřit jejich čas a celkově tak zrychlit chod zavedeného systému. Webová aplikace však byla vyvíjena poměrně rychle a to především proto, že se pro její pořízení rozhodlo vedení ve chvíli, kdy začala společnost rozšiřovat svoji působnost. Urychlený vývoj, nedostatečné znalosti analýzy a nesprávná dekompozice problémů v jednotlivých celcích přispěly k tomu, že program postrádá vnitřní logiku, jeho zdrojový text se v určitých pasážích stává zcela nepřehledným a obecně vzato je aplikace jen obtížně rozšiřitelná. Obsahuje bezpečnostní mezery, její databáze není efektivně koncipována a především pak postrádá spoustu funkcí, které by mohly uživateli usnadnit práci.

Cílem této práce tedy bude vytvořit nový, flexibilnější a snadněji rozšiřitelný systém doplněný o řadu dalších možností, jež současný uživatel postrádá. Jedním z aspektů, kterému se bude nová verze obzvlášť věnovat, je uživatelské rozhraní. Aktuální verzi mohl používat jen uživatel naprosto srozuměný se zavedeným systémem. Nově by mělo být ovládání aplikace přívětivé pro kohokoliv, kdo ji začne používat, a to bez jakéhokoliv zaškolení.

2 Analýza původního systému

2.1 Programová struktura

2.1.1 (X)HTML

Všechny HTML součásti aplikace, dále uváděné také jako **Správce**, jsou specifikovány standardem HTML 4.01 Transitional, ačkoliv se mnohdy nejedná o validní dokument. Ve většině případů je nevalidita způsobena například chybějícím atributem `alt` u elementu ``, nesprávným vnořováním tagů do sebe nebo strukturou DOMu narušenou vložením nadbytečného uzavírajícího či otevírajícího tagu při generování obsahu PHP skriptem.

Sémantická úroveň HTML dokumentů dosahuje velmi nízké úrovně. I když se nejedná o projekt, kde by toto odvětví hrálo zásadní roli, chaotické uspořádání zdrojového textu ubírá jeho přehlednosti a celkové logice. Tag `
`, který se v zdrojových souborech vyskytuje poměrně často a absolutně postrádá sémantický význam, sice plní svoji funkci zalomení řádky, ale stejného efektu lze dosáhnout i použitím elementu `<p>` s odpovídajícím formátováním, přičemž zůstane zachováno rozdělení obsahu do logických celků.

2.1.2 Kaskádové styly

Původní uživatelské rozhraní aplikace bylo navrženo tak, aby byl obsah aktuálně prohlíženého dokumentu načten rychle a bez zbytečných prvků, proto **Správce** až na malé výjimky nepoužívá ve svém layoutu obrázky a jeho CSS šablona je velmi stručná. Stylování jednotlivých stránek mění jen použitá písma, barvy, případně rozměry a tvary prvků.

Design aplikace využívá nové možnosti, totiž nastavení kulatých rohů. Primárně lze tohoto efektu dosáhnout CSS vlastností `border-radius`, která ale není podporována všemi prohlížeči (nebo jejich staršími verzemi). Šablona stylu si tedy žádá přinejmenším doplnění dalších adekvátních vlastností tak, aby byl obsah dokumentů optimalizován pro všechny prohlížeče (v maximální možné míře).

2.1.3 JavaScript a jQuery

JavaScript má v celé aplikaci dvě jednoznačné úlohy – obsluhu a kontrolu formulářů, jež jsou ve většině případů technologií AJAX odesílány ke zpracování na server, a realizaci některých vizuálních efektů a přidané klávesové podpory.

Aby byla manipulace s prvky DOMu a obecně celým HTML dokumentem jednodušší, optimalizovaná pro všechny prohlížeče a snáze zapsatelná, využívají skripty aplikace knihovnu jQuery rozšířenou o framework jQuery UI, jež připojuje nadstandardní metody pro práci s barvami či podporu drag&drop využívanou při sestavování tras řidičů.

I přes použití těchto doplňků je zdrojový text neorganizovaný a obsahuje rozsáhlé funkce o několika desítkách řádek s opakujícími se příkazy – hlavní soubor JavaScriptu je ukázkou toho, že před zahájením prací na původní verzi nebyla provedena žádná analýza a že aplikace byla psána rychle a neefektivně.

Mimo formální nedostatky obsahují skripty i mezery věcné. Obzvláště zanedbané je ošetřování chyb AJAXu při spojení se serverem. Ačkoliv aplikace implementuje základní ošetření chyb (např. při HTTP stavovém kódu 500), dojde-li k automatickému odhlášení uživatele v důsledku vypršení platnosti sezení, není schopna na tuto skutečnost jakkoliv zareagovat. Pokud uživatel nepřejde do jiné části **Správce**, nedozví se o automatickém odhlášení a aplikace se k němu bude i nadále chovat jako k přihlášenému. Při zadávání objednávky se tedy požadavek na zapsání dat do databáze odešle, ale na serveru se neprovede, což má fatální následky.

2.1.4 PHP a mPDF

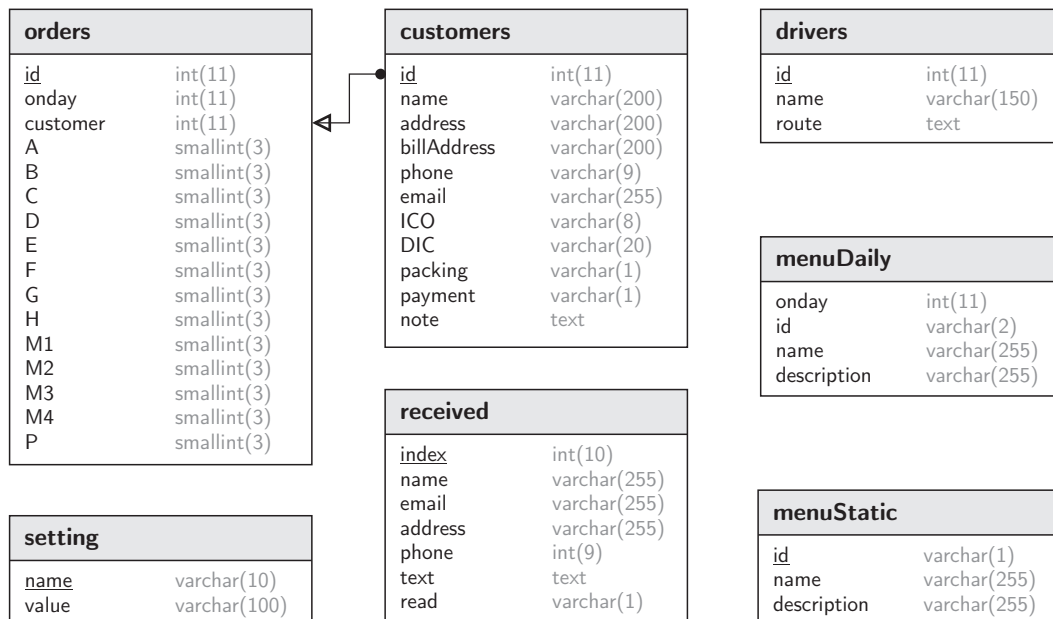
PHP skripty obsluhují v aplikaci veškerou dynamiku – komunikují s databází, zpracovávají získané výsledky, generují HTML šablony a PDF výstupy. Vzhledem k tomu, že ve **Správci** není využíváno objektového programování, působí PHP skripty zcela chaoticky, což umocňuje i fakt, že naprosto postrádají komentáře.

Vytváření šablon zajišťují pouze dvě funkce, které pomocí příkazu `echo` posílají na výstupní buffer HTML popis dokumentu. Při generování tedy nemá programátor příliš mnoho možností, jak vykreslení layoutu ovlivnit. I když se vzhled aplikace ve většině případů nemění, může být občas potřeba deaktivovat některé menu nebo skrýt určitý prvek standardní šablony, což v tomto případě nelze.

Aplikace umožňuje vytvářet 8 různých sestav připravených k tisku nebo archivaci. Ze **Správce** se odešle požadavek na konkrétní skript, jemuž je metodou GET předána informace o tom, který výpis uživatel vyžaduje. Samotné vytvoření PDF souboru zajišťuje knihovna mPDF, jež autor poskytuje pod GNU licencí a které je předán HTML výstup sestavy. Ten mPDF použije jako předlohu pro generování PDF dokumentu. Právě takové zpracování ale zřejmě představuje příčinu vyčerpání poskytovatelem stanoveného `memory_limit` (160 MB). K němu dochází především při generování týdenního záznamového archu řidiče, který obsahuje rozsáhlou tabulku se seznamem všech zákazníků v dané trase a odpovídajících polí pro zanesení počtů objednaných pokrmů na celý týden.

2.1.5 MySQL

Databáze aplikace vyžaduje kompletní přepracování, a to ať už kvůli formálním nedostatkům, tak kvůli jejímu špatnému návrhu. Z obrázku 2.1 (str. 4) je patrné, že byla vytvářena stejně jako ostatní součásti **Správce** bez jakékoliv předchozí analýzy. Jednotlivé tabulky jsou tedy provázány jen minimálně, neodpovídají přesně reálným modelům a nepoužívají efektivní principy.



Obrázek 2.1: ERA model původní databáze

Formální stránka databáze Tabulky v databázi by měly dodržovat určité konvence jejich názvů, pojmenování sloupců, indexů apod. Toto pravidlo databáze aplikace vůbec nedodržuje. Některé tabulky jsou pojmenovány názvem objektu, který reprezentují, v množném čísle a jiné v jednotném (**order** a **customers**). Stejně tak se název sloupce s indexem napříč tabulkami liší (**id** a **index**).

Primární klíč Tabulky **setting** a **menuDaily** nemají definovaný primární klíč, přestože právě ten slouží k jednoznačné identifikaci každé položky tabulky [1].

Datové typy Velké množství sloupců používá pro uchovávání hodnot neodpovídající datové typy nebo typy se zbytečně velkým rozsahem. Příkladem může být použití **INT(11)** pro uložení data (např. sloupec **onday** v tabulce **order**), což vyžaduje 4 bajty paměti. Není ani nastaven atribut **UNSIGNED**, takže se využívá pouze polovina rozsahu (časový otisk nemůže být záporný). Proto je v tomto případě mnohem vhodnější použít datový typ **DATE**, který pro uchovávání takové informace slouží, a navíc pro něj vyžaduje pouze 3 bajty.

Podobně je tomu také u sloupce **phone**, jenž se vyskytuje v tabulkách **received** a **customers** a uchovává devítimístné telefonní číslo. V každé tabulce je ale definován jinak – jednou jako **VARCHAR(9)** a podruhé **INT(9)**. Paměťově úspornější je samozřejmě varianta, kdy se telefon uchovává jako číslo, a vyžaduje tedy pouze 4 bajty. Druhá možnost, při níž se telefon považuje za řetězec o 9 znacích, pro své uložení potřebuje 10 bajtů (9 pro znaky, desátý pro uložení délky).

Nevhodné ukládání tras Trasy řidičů se skládají z jednotlivých zákazníků, které musí obsloužit. V databázi jsou všechny obsluhované osoby uloženy jako řetězec jejich ID oddělených znakem „|“, trasa je tedy reprezentována jako jedna hodnota ve sloupci `route`. To samozřejmě porušuje principy 1. normální formy, podle níž mají být atributy atomické [1], a zároveň tedy i znemožňuje přímé propojení tabulek `customers` a `drivers` pomocí klauzule `JOIN`. Navíc si změna trasy pokaždé vyžádá kompletní vygenerování nového řetězce.

2.2 Bezpečnost

2.2.1 Zabezpečení přenosu dat

Aplikace nepoužívá protokol se šifrováním HTTPS. Poskytovatel hostingu [5], na němž je aplikace provozována, sice nabízí podporu HTTPS, ale neposkytuje potřebný certifikát. Koncový uživatel aplikace byl o případných dopadech informován a souhlasil s použitím nezabezpečeného protokolu HTTP.

2.2.2 Ochrana hesel

Aktuální verze `Správce` pracuje pouze s jediným heslem, a tím je přístupové heslo. Protože k aplikaci doposud přistupoval jediný uživatel, bylo při uvedení programu do provozu stanoveno pevné heslo, jež se skrývá přímo ve zdrojovém textu v podobě MD5 otisku. To sice znemožňuje jeho odhalení, ale také nelze toto heslo měnit bez zásahu do zdrojového textu.

2.2.3 SQL a PHP injection

SQL injection je způsob útoku, kdy se do SQL dotazu vloží prostřednictvím uživatelského vstupu posloupnost SQL příkazů. Pokud vývojář s takovou situací nepočítá, dojde k úpravě SQL dotazu do formy, která může útočníkovi poskytnout citlivé údaje z databáze. Uvažujme situaci, kdy vývojář odesílá do databáze pomocí PHP dotaz:

```
mysql_query("SELECT heslo FROM uzivatel  
WHERE uzivatel.jmeno = '" . $_POST["login"] . "'");
```

Podstrčí-li uživatel jako vstup v poli `login` řetězec `někdo' OR '1' = '1`, bude ve výsledku databázi zaslán dotaz

```
SELECT heslo FROM uzivatel  
WHERE uzivatel.jmeno = 'někdo' OR '1' = '1'
```

Vybrány tedy budou všechny záznamy, protože výraz `'1' = '1'` je vždy pravdivý [6]. Takovému chování se dá zabránit adekvátním escapováním vybraných znaků (např. znaku apostrofu), které pak ztratí svůj význam v SQL dotazu a jsou

chápaný jako část řetězce. Pro tyto účely slouží tzv. sanitální metody, mezi něž se řadí např. `mysql_real_escape_string` [3]. Správce však užití této ochrany na-prosto opomíjí a potenciálnímu útočníkovi tak usnadňuje případný útok.

Obdobný problém představuje PHP injection. Vývojář načítá pomocí funkce `include` externí PHP skript, jehož název zmíněné funkci předává hodnotou z pole `$_GET`. Např.:

```
http://stranky.cz/index.php?page=a.html
```

Útočníkovi pak stačí vytvořit skript s funkcí `show_source` [2], jež zobrazuje zdrojový text PHP skriptu s předaným názvem, umístit jej na nějaký hosting a pak jen předat jeho URL adresu jako hodnotu proměnné `$_GET["page"]` [7].

Ve stávající verzi Správce nemůže k PHP injection dojít, protože není nikde ve zdrojovém textu použito volání funkce `include` s proměnným argumentem. Nelze však vyloučit SQL injection, protože vůči takovému útoku není aplikace chráněna a vyžaduje si tedy v tomto směru značné úpravy.

2.3 Funkční celky aplikace

2.3.1 Výpisy

Správce umožňuje uživateli převádět vybrané záznamy z databáze do „lidské“, a tedy i tisknutelné podoby, resp. do přenositelného formátu PDF. Data a způsob jejich formátování pro tisk stanovuje interně aplikace, a proto je nemožné výpis jakkoli přizpůsobovat. Uživatel může jediné ovlivnit specifikaci výběru dat – časové rozmezí, řidič, zákazník apod.

Stávající aplikace nabízí osm vestavěných výpisů, přičemž jedinou možností, jak tento počet změnit, je zásah do zdrojového textu aplikace.

2.3.2 Příchozí objednávky

Tato volba nebyla v prvotní verzi programu obsažena, protože provozovatel zařízení přijímal objednávky na pracovní e-mail. K problémům došlo ve chvíli, kdy se elektronické zprávy generované e-shopem na stránkách společnosti přestaly do schránky zasílat. Abychom se takové situaci do budoucna vyhnuli, společně se zasílanou zprávou se do databáze začal ukládat záznam o přijetí nové objednávky.

V menu Příchozí má pak uživatel kontrolu nad objednávkami přijatými přes e-shop v posledních 14 dnech. Nepřečtené položky jsou označeny červeně a po klepnutí na jméno zákazníka se zobrazuje kompletní objednávka. Uživateli se nabízí také otevření e-mailové adresy objednávacího výchozím poštovním klientem.

2.3.3 Řidiči

Správce nabízí skutečně málo možností pro správu řidičů. Jediným údajem, který řidič má, je jméno. Menu Řidiči tedy slouží pouze pro jejich editaci nebo vytvoření/odstranění stávajícího řidiče.

2.3.4 Zákazníci

Menu Zákazníci nabízí uživateli aplikace přidávání či editaci zákazníků. Každý nový zákazník může být vytvořen vyplněním formuláře, který se permanentně nachází v levé části obrazovky. Vpravo je pak umístěn seznam abecedně uspořádaných zákazníků. S přibývajícím počtem osob v adresáři však přestává být přehledný, protože uživatel nemá žádné prostředky k jeho filtrování.

2.3.5 Trasy

Každý řidič má daný seznam zákazníků, které musí každý den obsloužit. Jejich pořadí a množství lze editovat právě tímto menu. Uživatel zvolí řidiče, jehož trasu chce upravit, a aplikace mu nabídne pohled na aktuální trasu a seznam zákazníků, kteří nejsou zařazeni k žádnému řidiči. Tažením uživatel provádí změny. Pro dokončení musí klepnout na tlačítko Uložit – mnohdy tak neučiní, protože na tento krok zapomene, a zapříčiní tak ztrátu provedených změn.

2.3.6 Objednávky

Uživatel potřebuje každý den komplexně upravovat objednávky všech zákazníků, k čemuž mu poskytuje odpovídající možnosti toto menu. Po výběru dne a řidiče se načte trasa a počty objednaných pokrmů u jednotlivých zákazníků. Tyto počty může uživatel přímo editovat a jejich změna se okamžitě ukládá. Proces zadávání objednávek totiž probíhá u všech řidičů každý den a dohromady může trvat až 2 hodiny, proto je nežádoucí, aby došlo k případné ztrátě dat způsobené nepravdělným ukládáním změn.

2.3.7 Nastavení

V současné verzi Správce nemá uživatel příliš mnoho možností personalizace či jakéhokoliv jiného přizpůsobení aplikace. Jedinými dostupnými volbami je nedostatečná editace ceníku pokrmů, jež uchovává tři cenové hodnoty, a vytváření zálohy.

Aktuální způsob nastavování ceny se však vůbec nehodí. Pokud uživatel cenu změní, změní se i cena již prodaných obědů a některé výpisy (např. měsíční výpis) pak vůbec neodpovídají skutečnosti.

2.4 Uživatelské rozhraní

Doposud používal aplikaci jediný uživatel, jenž byl zaškolen a který s ní pracuje každý den. Avšak v případě, kdy by došlo k indispozici současného uživatele, musela by s aplikací pracovat nová, nezaškolená osoba. S takovou situací **Správce** nepočítá, a to je právě patrné i z navrženého uživatelského rozhraní. Tato skutečnost se mnohdy projevuje v drobných detailech.

Svoji roli zde opět hrála také nedostatečná předchozí analýza – vývojář nebyl plně seznámen se systémem zavedeným ve společnosti, postrádal podrobnosti, a sběr informací tedy nebyl dostatečný. Jak text již jednou uváděl, program byl vyvíjen rychle a důraz byl kladen především na jeho praktickou stránku a snadný běh. Uživatelská přívětivost se tedy přesunula do pozadí. Nedostatky uživatelského rozhraní **Správce** podrobněji popisuje několik následujících podkapitol.

2.4.1 Globální aspekty

Dostupnost hlavního menu

Protože se jedná o webovou aplikaci, není vertikální rozměr HTML dokumentu nikterak omezen. Proto může uživatel obsah většiny částí programu rolovat pomocí posuvníků. Vzhledem k tomu, že v aplikaci ale neexistují žádné fixně pozicované elementy, odroluje se spolu s aktuálním obsahem také hlavní nabídka. Pro přechod do některého jiného funkčního celku pak musí opět rolovat obsah zpět, aby mu bylo zobrazeno hlavní menu. To může zpomalovat práci s aplikací.

Prvky prohlížeče

Některé ovládací prvky prohlížeče mohou v uživateli vyvolat dojem, že fungují stejně jako při klasickém procházení webových stránek, což ovšem nemusí být vždy pravda. Například tlačítko **Zpět** mnohdy neumožňuje skutečný krok navracení v historii tak, jak by uživatel očekával. Některé funkce jsou totiž obsluhovány pomocí technologie AJAX, která webovou historii vůbec neovlivňuje.

Absence upozornění při navigaci

Stávající aplikace neupozorňuje uživatele při přechodu do jiného funkčního celku (na jinou fyzickou stránku), takže může poměrně snadno dojít ke ztrátě prováděných změn, například při editaci tras řidičů. Klepne-li uživatel na tlačítko **Zpět** ve svém prohlížeči, nedokáže na to **Správce** adekvátně zareagovat třeba zobrazením dotazu na uložení změn.

2.4.2 Ovládací prvky

Volba data a datového rozmezí

Doposud měl uživatel možnost volit datum pouze prostřednictvím výběrového pole. Jeho výchozí hodnota byla zpravidla nastavena na aktuální den či jiný den odpovídající účelu výběru (např. při volbě týdne obsahovalo jedno pole hodnotu prvního a druhé pole hodnotu posledního dne v tomto týdnu, viz obr. 2.2). Zcela nepřívětivá situace tedy nastala v okamžiku, kdy chtěl uživatel změnit třeba již zmíněné časové rozmezí. V jednom výběrovém poli musel vyhledat nový počátek týdne a podobně změnit i hodnotu druhého prvku.

Obecně vzato je výběr data, jenž se v aplikaci vyskytuje poměrně často, řešen nešťastně. Většina uživatelů očekává při volbě data zobrazení kalendáře či jiného pomocného prvku, který jim usnadní orientaci např. zobrazením po týdnech nebo zvýrazněním aktuálního dne.

Obrázek 2.2: Volba datového rozmezí

	P	M1	M2	M3	M4	A	B	C
COOP HOSTOŮŇ	0	0	0	0	0	0	0	0
AGRAZET JZD	1	0	3	0	0	0	0	0
SP PLATÍ ****5 KČ*** VAK	0	2	3	0	0	0	0	0
POBĚŽOVICE VÁRNICE	32	16	11	5	0	0	0	0
POBĚŽOVICE VAX	3	2	0	1	0	0	0	0
35	0	1	0	0	0	0	0	0

Obrázek 2.3: Tabulka objednávek

Tabulka objednávek

Tabulka objednávek se skládá z množiny textových polí (obr. 2.3), jejichž hodnota informuje o počtu objednaných pokrmů a lze také okamžitě změnit (včetně uložení

změn). Prvek má pevně stanovenou výšku, aby při změně řidiče, jehož objednávky jsou zaznamenávány, resp. prohlíženy, nedocházelo ke změnám rozměrů celého HTML dokumentu. Z toho plyne, že prvek musí být umístěn uvnitř posuvné vrstvy. To si také vyžaduje fixní záhlaví, aby při rolování obsahu tabulky nedošlo k jeho skrytí.

Uživatel má možnost používat pro pohyb mezi vstupními poli kurzorové klávesy, Enter a klávesy A–F, jež odpovídají označení pokrmů ze stálé nabídky (ty ovšem nejsou vůbec využívány). To s sebou přináší problémy právě při rolování obsahu prvku – není-li pole, jež získalo fokus, viditelné (resp. vyžaduje posun ke svému zobrazení), některé prohlížeče automaticky nutný úsek odrolují tak, aby daný prvek zobrazily. Avšak ne všechny a ne vždy. Navíc odrolují skutečně jen bezpodmínečně nutnou část a cílový prvek tedy zobrazí zcela u spodního okraje prvku rodičovského.

Ačkoliv má uživatelovu orientaci v tabulce usnadnit fixní záhlaví, ne vždy se mu to podaří. Obsluhující osoba mnohdy potřebuje vědět, který oběd se skrývá pod označením M1 v aktuální den. Aby to zjistil, musí v současné verzi Správce přejít do zcela odlišného menu, což celý proces zaznamenávání objednávek rapidně zdržuje a odvádí uživatelovu pozornost. Tím se samozřejmě zvyšuje riziko chyby lidského faktoru.

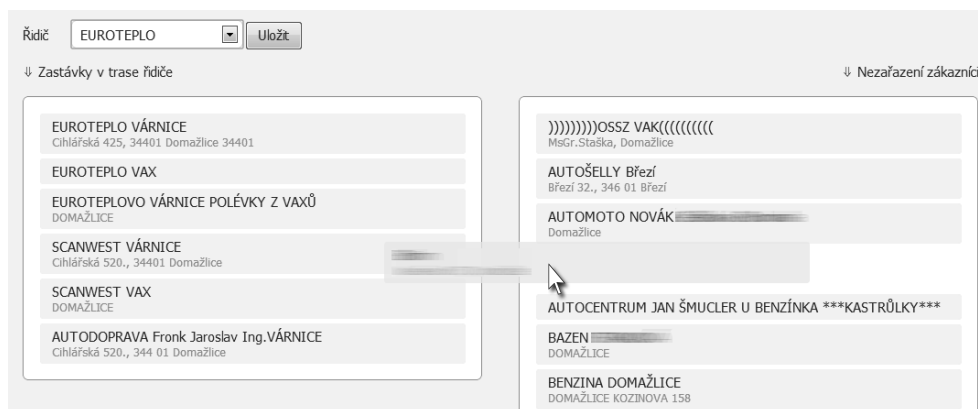
Vzhledem k tomu, že prvky tabulky mají mimo funkci informativní také editační, může uživatel libovolně měnit jejich obsah, přičemž ztráta fokusu prvku vyvolá funkci pro uložení. Při ukládání může dojít k chybám – o tom je uživatel informován zbarvením daného pole do červena. V opačném případě se pozadí editovaného prvku změní na zelenou barvu. Obsluhující osoba se už v případě chyby ale nedozví, co ji způsobilo/způsobuje.

Editor trasy

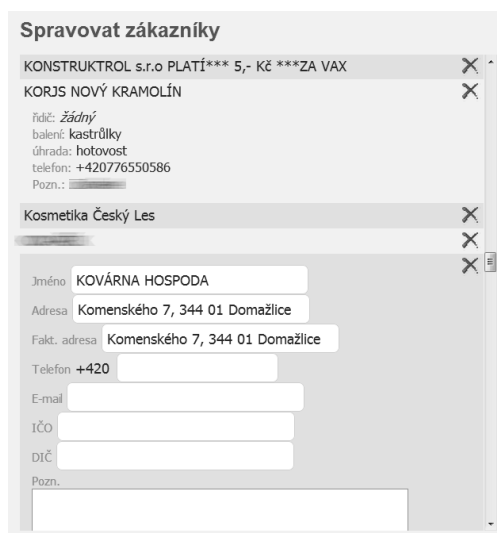
Editace trasy je realizována dvěma seznamy (viz obr. 2.4) – jeden zobrazuje aktuální trasu řidiče a druhý zákazníky, kteří nejsou zahrnuti v žádné z tras. Pouhým přetažením lze editovat pořadí položek trasy a stejně tak lze mezi oběma seznamy jednotlivé zákazníky přetahovat. Ve chvíli, kdy je jeden ze seznamů delší než okno prohlížeče, začne být přetahování velmi nepohodlné. Uživatel musí přetahovanou položku podržet u dolního okraje okna prohlížeče a čekat, až se odroluje potřebná část obsahu, aby mohl zákazníka umístit na adekvátní pozici.

Adresář zákazníků

Pro zobrazování zákazníků je vyhrazena pouze polovina stránky v menu Zákazníci. K zobrazení všech detailů u jednotlivých položek adresáře je tedy k dispozici skutečně málo prostoru, obzvláště protože v jednom prvku lze položky i editovat (obr. 2.5). Zákazníky není možné jakkoliv filtrovat, takže je obsluhující osoba nucena vždy procházet celý seznam. Editace je realizována tak, že po klepnutí na odpovídající tlačítko se rozbalí formulář pro úpravu údajů přímo v adresáři.



Obrázek 2.4: Editor tras při přesouvání zákazníka



Obrázek 2.5: Prohlížení a editace zákazníků

3 Návrh nového systému

3.1 Obecná charakteristika

3.1.1 Rozsah projektu

Vyvíjený systém bude doplněn o spoustu nových funkcí a také ty stávající projdou množstvím změn. Z toho plyne, že rozsah projektu se už jen díky těmto faktorům několikrát zvětší oproti systému používanému v současnosti. Kompletní podobu finálního produktu popisuje diagram funkčních celků, viz příloha A. Protože se ale jedná o rozsáhlou aplikaci, v rámci této bakalářské práce implementují pouze základní funkcionalitu, přičemž v práci na dokončení systému budou pokračovat i po obhajobě. V souladu se zadáním implementují:

- uživatelské rozhraní, tj. nakódují grafický layout aplikace,
- autorizaci a autentizaci uživatele,
- správu zákazníků, objednávek a distribuce.

3.1.2 Kontext systému

Jak bylo zmíněno hned v úvodu, administrativní pracovnice stravovacího zařízení momentálně aplikaci pro správu používá, avšak důvody uvedené v kapitole 2 znemožňují její snadnou obsluhu, a navíc program postrádá některé funkce. Nově vyvíjený systém, který již bude možné považovat za plnohodnotný softwarový produkt, tedy nahradí stávající aplikaci v plném rozsahu, přičemž zůstane zachováno jeho fyzické umístění.

3.1.3 Funkce produktu

Protože se jedná o náhradu existujícího systému, zůstanou samozřejmě zachovány jeho původní funkce, tj. správa objednávek, zákazníků a řidičů. Navíc ale bude rozšířen o mnoho úprav a dalších možností (kompletní přehled popisuje diagram, viz příloha A). Nová aplikace uživateli poskytne také následující funkce:

- *nástěnka*, která nabídne plochu pro „připíchnutí“ lístků s poznámkami a okamžitý přístup k vyhledávání v databázi;
- *finance*, pomocí nichž může uživatel vést peněžní deník, spravovat pohledávky a vytvářet faktury (doposud muselo zařízení využívat dalšího programu, a uchovávat tak adresy a jména zákazníků duplicitně);
- *stravování*, jež nabídne nástroje pro správu připravovaných pokrmů, zobrazí statistiky jejich oblíbenosti a umožní sestavovat a hromadně rozesílat týdenní jídelníčky.

3.1.4 Autentizace a autorizace

Autentizace slouží k jednoznačné identifikaci uživatele, který chce vstoupit do systému [4]. V původním systému byla realizována tak, že jakékoliv osobě stačilo znát přístupové heslo a mohl se do systému přihlásit – nebyl tedy identifikován jednoznačně. Tehdejšími požadavky zadavatele tato alternativa vyhovovala, jelikož systém spravovali maximálně dva uživatelé. Nová verze aplikace umožní vytvářet uživatelské účty, čímž zajistí jedinečnou identifikaci každé přistupující osoby.

Autorizace pracuje s již autentizovaným uživatelem a rozhoduje o tom, jaká práva jsou mu v systému přidělena [4]. Aplikace používaná v současnosti proces autorizace zcela vynechává, logicky to plyne ze skutečnosti, že se v systému může vyskytovat jediná identita. Oproti ní nově navržený systém s autorizací počítá, realizována bude pomocí ACL (Access Control List), který představuje sbírku omezení a přístupových pravidel pro jednotlivé uživatelské role.

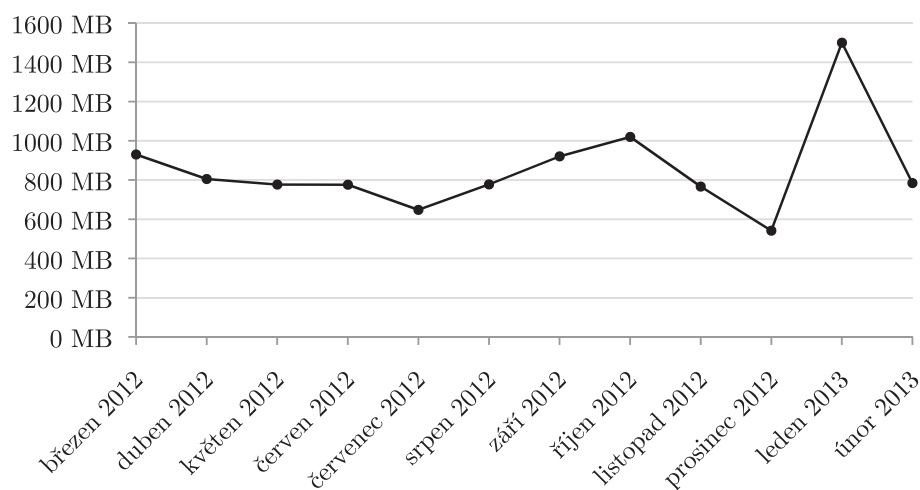
3.1.5 Provozní prostředí

Při konzultaci vývoje aktuálně užívané aplikace zadavatel uvedl, že by bylo vhodné, aby mohl do systému vstupovat jak v prostorách stravovacího zařízení, tak také z osobního počítače doma. Objednávky jsou sice zpravidla zadávány prostřednictvím počítače v kanceláři, avšak během týdne se shromažďují předběžné počty pokrmů na následující týden a ty je potřeba zanést do databáze o víkendu. Jedině tak je možné informovat na začátku nového týdne personál kuchyně o přibližném množství objednaných pokrmů, a pořídit tak odpovídající množství potřebných surovin.

Tyto důvody tedy vyžadovaly, aby program sdílel data přes Internet. Stěžejní skutečnost tedy představovalo rozhodnutí, zda koncipovat nový systém jako tenkého či tlustého klienta. V druhém případě by se nabízelo vytvořit aplikaci například v jazyce Java, který by pak komunikoval s databází umístěnou v síti. Tento způsob by však vyžadoval distribuci aplikace na všechny počítače, které by měly práci se systémem umožnit, což je typickým znakem tlustého klienta. Zvolil jsem tedy raději alternativu tenkého klienta, a to v podobě webové aplikace. Systém správy tak není nutné šířit na více počítačů, protože uživatel může k administrativnímu rozhraní přistupovat přes webový prohlížeč z jakéhokoliv počítače. Tato koncepce sice vyžaduje práci online, tj. trpí typickou nevýhodou tenkého klienta [8], ale i přesto zůstane zachována u nové verze programu, i nadále půjde o webovou aplikaci.

U nového systému se zdatelně rozšíří škála uživatelů, kteří k němu budou přistupovat. Spolu s rozvojem firmy totiž dochází k drobnějšímu rozdělení jednotlivých úkonů, a tak administrativní pracovnice spravuje zákazníky a přijímá pouze elektronické objednávky, zatímco jejich denní aktualizaci provádí sami řidiči.

I přes navýšení počtu uživatelů a případnou další distribuci, bude aplikace lokalizována pouze v českém jazyce. Využití plánovaných technologií však poměrně snadno dovolí případnou lokalizaci do jiného jazyka dodatečně provést.



Obrázek 3.1: Graf znázorňující objem přenesených dat v rámci webových stránek stravovací společnosti a jí používaným systémem správy (statistiky poskytovatele hostingových služeb)

3.1.6 Omezení návrhu a implementace

Samotný proces vývoje aplikace není nijak časově omezen, protože stravovací zařízení v současnosti používá původní verzi správce, která je plně funkční, ačkoliv v ní uživatel postrádá některé zmíněné funkce či jejich úpravy. Jak již bylo zmíněno v kapitole 3.1.1, v rámci této práce bude vytvořena základní kostra systému, na jejímž dalším vývoji budu intenzivně pokračovat i po obhajobě.

Implementaci omezují pouze parametry webhostingu, na němž je program provozován (endora.cz). Vzhledem k tomu, že zadavatel byl již dříve ochoten zakoupit hostingový program, který nabízí nejlepší možné vlastnosti, jsou omezení skutečně malá. Program Mega [5] (k 8. dubnu 2013)

- nabízí 100GB měsíční limit přenosu dat, jenž pravděpodobně nebude nikdy vyčerpán (viz obrázek 3.1),
- garantuje zálohu jak databáze, tak i souborů aplikace,
- poskytuje PHP `memory_limit` 160 MB a maximální dobu běhu skriptu 30 sekund,
- umožňuje rozeslat 400 emailů za hodinu,
- poskytuje MySQL databázi typu InnoDB s velikostí až 150 MB.

3.1.7 Systém jako softwarový produkt

Jelikož byla aktuálně používaná aplikace vyvíjena pro jedinou společnost, neuvažoval jsem nad jejím pojmenováním a pro označení používal obecného jména správce. Během analýzy a vývoje nového systému, kdy bylo potřeba konzultovat změny se zadavatelem, se běžné označení používalo jako název, a tak byla aplikace alternativně pojmenována **Správce**. Z hlediska případné propagace či distribuce systému jako softwarového produktu nepředstavuje toto označení vhodný název, proto jsem sestavil seznam možných alternativ a konzultoval finální výběr se zadavatelem a 5 nezávislými osobami. Nejvíce kladných ohlasů získalo pojmenování **Gadmin** tvořené spojením výrazů *gastro* a *administration*. K názvu byl vytvořen také logotyp, viz obr. 3.2.



Obrázek 3.2: Logotyp nového systému Gadmin

3.2 Inovace a nové funkce

Následující kapitoly detailněji popisují nové funkce systému, popř. úpravy, kterými se Gadmin od dříve užívané aplikace odlišuje. Kompletní návrh systému a provázanost mezi jednotlivými funkčními celky demonstruje diagram, viz příloha A.

3.2.1 Nástěnka

Podobně jako ve všech kancelářích i v administrativním prostředí stravovacího zařízení jsou pracovní stůl a zeď posety lístečky s informacemi různého charakteru. Ať už se jedná o zapsané telefonní číslo, dodatečnou objednávku či instrukce, jaké suroviny mají být objednány na další týden, mohou se ve velkém množství dalších papírů a poznámek poměrně snadno ztratit. Neblahým následkem pak může být nespokojenost cílového zákazníka, jenž např. neobdrží oběd. Právě tomuto „lístečkování“ má zamezit nová část systému – **Nástěnka**.

Uživatel systému může přidávat lístečky na virtuální nástěnku, která vylučuje jejich ztrátu. Prostřednictvím tohoto menu má navíc rychlý přístup k vyhledávání v databázi či vygenerování často generovaných výpisů.

Dostupné interakce

Lístičky mohou být přidávány, editovány nebo odebírány. Pokud uživatel neodebere lístek sám, po uplynutí nastaveného termínu, který je pro každou poznámku jiný, dojde k automatickému odstranění. Panel pro vyhledávání nabízí uživateli procházet databázi podle telefonních čísel, jmen, typu balení apod. Jeho koncepce se zaměřuje především na rychlý přístup k požadovaným datům např. během telefonního hovoru.

3.2.2 Zákazníci

Zákazníci mohou být dvojího druhu – fyzické osoby nebo společnosti. Doposud je systém nijak nerozlišoval, což uživatele aplikace značně omezovalo. Nemohl totiž k jednomu zákazníkovi připojit více e-mailových adres nebo telefonních čísel, protože databáze automaticky operovala se zákazníkem jako jednotlivcem. **Gadmin** nově nabídne u společností uvádět kontaktní osoby a s nimi i údaje tohoto charakteru.

Samotné menu **Zákazníci** se transformuje na regulérní adresář. Uživatel může zobrazit seznam všech zákazníků, řadit je a filtrovat nebo přejít na kartu vybraného zákazníka, a získat tak kompletní přehled o jeho posledních objednávkách a dalších informacích.

Se systémem **Gadmin** se také zcela mění způsob kalkulace cen objednávek. Dřívější aplikace zaváděla jednotnou cenu pro všechny objednávky. Protože ale některé větší společnosti mohou pro své zaměstnance získat množstevní slevu na stravování, má nyní administrativní pracovnice stravovny možnost změnit cenu pokrmu u každého zákazníka.

Dostupné interakce

Adresář zákazníků může uživatel filtrovat a řadit. Dále má možnost přidávat zákazníky, editovat jejich kontaktní a odběrové údaje (způsob balení, platby), skrývat je nebo jim zaslat e-mail přímo prostřednictvím aplikace. Zákazníka nelze trvale odstranit, protože je vázán na své objednávky z dřívějšího období. Z karty zákazníka lze pak přejít do menu **Finance** (viz kapitola 3.2.5) a zobrazit jeho pohledávky vůči stravovacímu zařízení.

3.2.3 Distribuce

Toto menu zůstává téměř v nezměněné podobě. Jediné výrazné odlišení od předchozího systému představuje rozdělení distribuce na řidiče a výdejny. Jedná se však pouze o terminologii, protože oba dva výdejní prvky se v rámci aplikace chovají stejně.

Dostupné interakce

Uživateli je poskytnuta možnost zakládat nové řidiče (resp. výdejny), upravovat jejich údaje a trasu. Editace trasy spočívá v přidávání/odebírání zákazníků nebo změně jejich pořadí. Podobně jako zákazníci (kapitola 3.2.2) i u řidičů a výdejen může obsluha aplikace zobrazit kartu s detaily.

3.2.4 Objednávky

Kvůli tomu, že samotná stavba databáze programu projde markantními změnami, se také systém objednávání zdatelně změnil. V současnosti se záznamy o objednaných pokrmech uchovávají pouze po dobu 90 dnů. Ukázalo se ale, že pro potřeby stravovacího zařízení je tento limit příliš omezující (někdy se vyžaduje přístup k objednávkám starším i více než půl roku), proto budou v novém systému objednávky zaznamenány permanentně.

Podobně jako v menu Zákazníci (kapitola 3.2.2) i u objednávky má nově uživatel možnost explicitně změnit cenu. Primární cena je reprezentována hodnotou uvedenou v nastavení ceníků, sekundárně ji může ovlivnit cena specifikovaná u jednotlivých zákazníků a při zadávání objednávky pak lze dvě zmíněné hodnoty nahradit přímo vloženou hodnotou. Tento systém zajistí, že výpisy, které nějakým způsobem operují s cenami pokrmů, budou vracet i v případě půl roku starých dat korektní výstup. Ve stávající aplikaci tomu tak nebylo – změna globální ceny totiž zapříčinila také změnu ve starých výpisech. Takové chování samozřejmě není žádoucí. Mimo tříúrovňové stanovení cen poskytne nový program možnost definovat také cenu obalového materiálu, která se automaticky připočte k celkové hodnotě objednávky.

Koncepce vkládání objednávek rovněž projde změnami. Jedním z požadavků zadavatele bylo, aby aplikace umožňovala vkládat objednávky pro celý týden najednou. Doposud mohl uživatel zadávat počty pokrmů jen na jeden den a poté musel přepnout na další.

Explicitní přijetí elektronické objednávky

Zákazníci stravovacího zařízení mají možnost objednat obědy prostřednictvím rozhraní na webových stránkách, pokaždé však musí zadat doručovací údaje, aby mohla administrativní pracovnice zákazníka bezproblémově identifikovat. Objedávka takového charakteru se pak automaticky odešle na e-mail společnosti, z něhož musí obsluha stravovny data přenést do systému správy. Proto byla současně se systémem **Gadmin** předložena koncepce zákaznického ID, jež by nutnost zadání kontaktních údajů zákazníkem a následný přesun do systému eliminovala.

V případě, kdy zákazník využije této funkce, zobrazí se spolu s příchozí objednávkou nově možnost jejího explicitního přijetí, tzn. automatické zařazení požadavku do správy objednávek k zákazníkovi s odpovídajícím identifikátorem.

Další dostupné interakce

Osoba obsluhující aplikaci může volit mezi způsoby, jakými bude počty objednaných pokrmů vkládat – buďto vybere jediný den, nebo zvolí zadávání objednávek na celý týden pro vybranou trasu. Pro úpravu vložených dat použije stejné prostředí. Prostřednictvím menu **Objednávky** má také přístup k nově přichozím požadavkům, jejich správě či případnému přesunu do databáze.

3.2.5 Finance

Menu **Finance** představuje zcela novou součást systému. Momentálně musí administrativní pracovníce pro vytváření faktur a vedení peněžního deníku používat aplikaci **ÚČTO**. Tento produkt však v prostředí stravovacího zařízení neslouží pro kompletní účetní správu, nýbrž je využíván pouze k vedení elektronických záznamů informujících o finančních pohybech ve stravovně atp. (oficiální účetnictví se provádí mimo toto zařízení). To s sebou však přináší určitou duplicitu dat – zákazníci vyžadující faktury musí být vedeni jak v systému správy, tak i ve zmíněném účetním programu. Jakoukoliv změnu v datech musí tedy uživatel provést v obou aplikacích. Navíc nepůsobí faktury vygenerované systémem **ÚČTO** nikterak reprezentativně. Tyto důvody vedly zadavatele k požadavku integrace podobné funkcionality do aplikace **Gadmin**.

Mimo správu faktur má nový systém umožnit také vedení peněžního deníku, tj. poskytnout rozhraní pro přidávání záznamů o příjmu a výdeji peněz. Další součástí menu představuje přehled pohledávek, jenž obsluhuje stravovny nabídne seznam uhrazených a neuhrazených obědů, umožní je filtrovat podle řidičů apod. Uživatel tak získá ucelený pohled na dlužné částky a stav kasy.

Dostupné interakce

Autorizace tímto menu je povolena jen daným uživatelským rolím, protože se jedná o citlivá data, k nimž nemohou přistupovat osoby ze všech uživatelských tříd. Pokud tedy využívá více osob té role, jež přístup dovoluje, mohou editovat pouze data, jež do databáze vložili.

Jak již bylo v této kapitole zmíněno, osoba obsluhující systém (s patřičným oprávněním) může vkládat záznamy o výdeji či příjmu hotovosti do peněžního deníku. Tyto položky lze kategorizovat do skupin (definovány v menu **Nastavení**, kapitola 3.2.8). **Gadmin** také uživateli poskytuje nástroje pro vytváření, editaci a prohlížení faktur. Hlavní funkce menu **Finance** doplňuje kalkulačka nominálních hodnot, která umožňuje snadno přepočítávat hotovost či stravovací poukazy v kase.

Poznámka: Toto menu nebude implementováno v rámci bakalářské práce.

3.2.6 Stravování

Jedná se o další zcela novou součást systému. Doposud byly jídelníčky uváděny na web pomocí redakčního systému. Ten však nijak nespolečně spolupracoval s aplikací pro správu, proto bylo nutné každý nově vytvořený jídelníček publikovaný na webu exportovat do systému. Tam byla data používána pro generování přehledů zákazníky zakoupených pokrmů.

Gadmin spojí tyto dvě oddělené jednotky do jedné, a nově tak nabídne úplnou správu pokrmů a jídelníčků přímo uvnitř systému. Obsluha stravovny tedy nebude muset zadávat stejná jídla opakovaně, protože systém je automaticky registruje a sám nabídne při dalším sestavování jídelníčku. Díky permanentnímu uchovávání objednávek (viz kapitola 3.2.4) navíc aplikace umožní zobrazení statistik oblíbenosti jednotlivých pokrmů.

Dostupné interakce

Uživatelé se zpřístupňují nástroje pro sestavování a procházení týdenních jídelníčků. Ty sestávají z registrovaných pokrmů, jež může obsluha systému taktéž editovat. Při vytváření nového jídelního lístku lze explicitně vložit nový pokrm bez nutnosti přechodu do menu správy pokrmů. Systém totiž zjistí, že v databázi se takový zatím nenachází, a proto jej automaticky vloží.

Mimo automatické vkládání pokrmů kontroluje systém také četnost jejich použití. Snadno se může stát, že vytváří-li uživatel jídelníček manuálně, nevědomky některé pokrmy opakuje častěji, čehož si zaručeně všimne cílový zákazník. Gadmin jej na skutečnost, že se chystá použít nedávno nabízený pokrm znovu, upozorní.

Další novinkou je hromadné rozeslání jídelníčků formou e-mailové PDF přílohy. Administrativní pracovnice doposud musela jídelní lístek vytvořený v aplikaci Microsoft Word rozesílat standardním způsobem na e-maily zákazníků, které však byly uchovávány ve formátu vCard (souborový formát pro výměnu osobních dat, elektronická podoba obchodní vizitky). Nově může uživatel e-maily odeslat jedním kliknutím, protože veškeré e-mailové adresy jsou uvedeny u zákazníků v databázi.

Poznámka: Toto menu nebude implementováno v rámci bakalářské práce.

3.2.7 Výpisy

Stejně jako původní systém i Gadmin umožní vytvářet na základě výstupu z databáze tisknutelné a přenositelné výstupy ve formátu PDF. Ty slouží například pro řidiče, personál kuchyně, obsluhu výdejny apod. Nově bude toto menu koncipováno jako správce instalovaných modulů výpisu. Pokud si tedy uživatel v budoucnu vyžádá nový výstup (např. tisk statistik), vývojář mu pouze dodá soubor modulu a po instalaci jej bude možné ihned používat. Zásadní rozdíl tedy

spočívá v tom, že každý modul bude v novém systému umístěn do vlastního souboru, nikoliv do jednoho společného. Dostupné interakce samozřejmě závisí na zvoleném modulu. Standardně budou k dispozici následující:

- adresář s kontakty,
- přehled objednávek zákazníka,
- přehled pohledávek pro řidiče (tabulka se seznamem zákazníků na trase řidiče a přehledem částek, jež mají uhradit),
- záznamový arch (prázdná tabulka se zákazníky na trase řidiče a buňkami pro zanesení objednávek na nový týden),
- tisk přijaté objednávky,
- měsíční přehled,
- denní výpis pro kuchyni (obsahuje počty pokrmů filtrované dle tras řidičů a způsobu balení),
- tisk faktury,
- tisk jídelníčku,
- tisk výčetky (výstup z kalkulačky nominálních hodnot)

Poznámka: Toto menu nebude implementováno v rámci bakalářské práce.

3.2.8 Nastavení

Oproti stávající verzi systému poskytne **Gadmin** široké možnosti personalizace a dalšího nastavení systému. Celé menu bude rozděleno na podmenu, na něž se detailněji zaměřují následující kapitoly.

Personalizace systému

Jelikož nový systém poskytuje širší možnosti než původní aplikace, objevuje se i větší množství přizpůsobitelných prvků. Navíc se nově zavádí uživatelské role, a tedy se také počítá s větším počtem uživatelů programu. Ne každý bude chtít sdílet stejné nastavení jako ostatní, a proto dostane možnost změnit pomocí tohoto menu přinejmenším níže uvedené atributy uživatelského účtu:

- podpis – jméno, pod nímž uživatel v systému vystupuje (nikoliv uživatelské jméno),
- heslo a e-mail,
- zobrazení tipů – vypnutí/zapnutí zobrazování stručných nápověd.

Zálohování

Ačkoliv původní systém určitým způsobem zálohování podporoval, neumožňoval definovat, jak často a jakým způsobem má být na vytvoření zálohy uživatel upozorňován. Gadmin nabídne nastavení těchto atributů (autorizovaným uživatelům), a navíc umožní obnovu ze zálohy nahráním odpovídajícího souboru pro případ, kdy by došlo k selhání databáze.

Stravování a distribuce

Předmětem tohoto podmenu je přizpůsobení samotného menu **Stravování** a jeho výstupů. Stravovací zařízení používá k jednoznačné identifikaci speciální označení obědů (např. M1 představuje pokrm denního menu č. 1, B označuje pokrm stálého menu č. 2). Počet těchto označení se zpravidla nemění, nicméně se nabízí uživateli umožnit správu takových symbolů a neimplementovat je interně. Podmenu tedy uživateli poskytne následující možnosti:

- přidávání/úprava označení (zmíněno výše),
- přidávání/úprava typů balení,
- prodleva opakování – časový údaj definující, s jakým minimálním odstupem se mohou pokrmy v jídelnících opakovat,
- aktivace automatického ukládání pokrmů – volba, která umožní nově použitá jídla automaticky zanést do správy pokrmů,
- rozesílat jídelníčky kontaktním osobám – nastaví, zda se při hromadném odeslání nového jídelníčku mají e-maily pro firemní zákazníky směřovat na adresu celé společnosti nebo na adresy jejích kontaktních osob.

Finance a ceníky

Současný systém umožňoval pouze jednoduché nastavení cen, tj. definici ceny samotné polévky, samotného hlavního jídla a zvýhodněného kompletního menu. Během téměř dvouletého provozu se však měnily obchodní principy zavedené ve stravovacím zařízení, a tak nyní zadavatel vyžaduje zavedení dalších cen a tříúrovňové cenové hierarchie.

Primární cenu pokrmu stanovuje hodnota nastavená v tomto menu, sekundární tvoří údaj uvedený u jednotlivých zákazníků a jako terciární lze použít hodnotu uvedenou přímo při vytvoření objednávky.

Další součástí menu **Finance a ceníky** představuje editace skupin pro kategorizaci záznamů peněžního deníku, nastavení kalkulačky nominálních hodnot a definice formátování faktur. Obsluha aplikace bude mít k dispozici volbu, jež určí, zda má být na faktuře zobrazen název banky atp. Důležitým nastavením, jež menu **Finance a ceníky** poskytne, je editace daní využívaných ve fakturách či peněžním deníku.

Kalendář

Speciálním přáním zadavatele bylo, aby jej aplikace upozorňovala na narozeniny a svátky zaměstnanců stravovacího zařízení. K tomuto účelu slouží právě kalendář, kam může k jednotlivým dnům přiřadit upozornění. Kalendář navíc automaticky integruje české státní svátky a jmeniny a v budoucnu pravděpodobně dovolí také zaznamenávat termíny zaměstnaneckých dovolených.

Výpisy

Protože u předchozí verze systému zadavatel někdy potřeboval určité části z daného výpisu odstranit nebo je zobrazit jinak, vzniklo také toto podmenu, které umožňuje globálně pracovat s dokumenty výpisů. Obsluha aplikace může definovat, jaký text se má zobrazovat v záhlaví, zápatí, jaké jsou rozměry stránky a odsazení od jejího okraje a především pak, zda má být výstup ve formátu PDF či HTML.

3.3 Koncepte uživatelského rozhraní

Doposud aplikaci používala pouze jediná osoba – administrativní pracovnice stravovacího zařízení. Při vývoji první verze systému byly tedy brány v úvahu pouze její připomínky a vlastní návrhy vývojáře. Již několikrát zmiňovaná nedostatečná analýza zavedených postupů v tomto zařízení zapříčinila vznik nedostatků, a to nejen v oblasti funkčnosti, ale také v rámci použitelnosti a přívětivosti uživatelského rozhraní.

Nový systém tedy musí nabídnout jednoduchý a přehledný způsob správy společnosti tak, aby jej mohl bez větších problémů obsluhovat i uživatel, jenž nebyl kompletně s aplikací Gadmin seznámen. Následující odstavce stručně popisují změny v jednotlivých oblastech rozhraní systému či nově použité ovládací prvky.

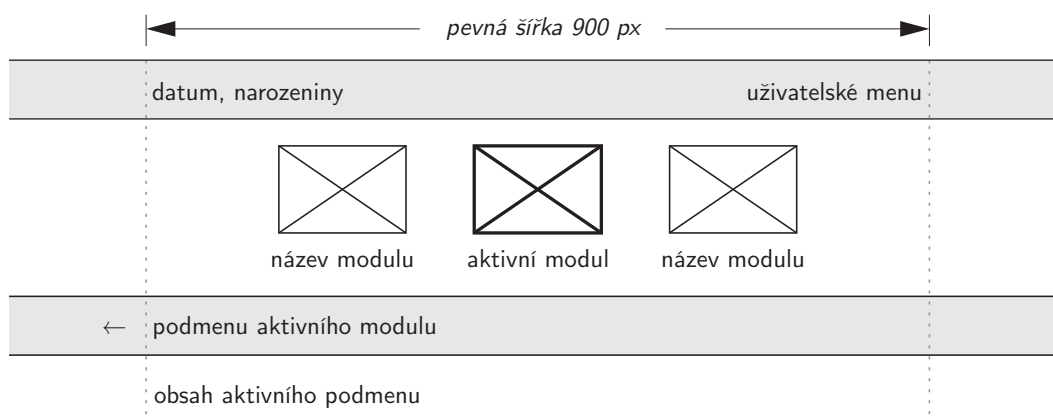
Hlavní panel

Hlavní panel původního programu zcela zbytečně zabíral spoustu prostoru v horní části HTML dokumentu. Ten však mohl být využit pro zobrazení podstatnějších prvků systému. Součástí panelu představovalo také hlavní, jednoúrovňové menu. Systém Gadmin jej dekomponuje na tři části (obr. 3.3):

- hlavní nabídku, kde uživatel zvolí modul, s nímž chce pracovat (např. **Zákazníci** nebo **Stravování**);
- podnabídku, jejíž obsah závisí na volbě modulu; vždy však obsahuje tlačítko **Zpět** pro návrat do předchozího menu;
- menu uživatele, které slouží pro personalizaci systému a k odhlášení z účtu.

Důležitou změnou je také zvýraznění aktuálně vybraného modulu. Doposud totiž uživatele systém nijak neinformoval o tom, ve které jeho části se nachází.

Hlavní panel dále doplňuje navigační lišta umístěná na konci každé stránky, jež obsahuje pouze dvě tlačítka pro rychlejší navigaci – Zpět a Nahoru.



Obrázek 3.3: Návrh horního panelu systému Gadmin

Větší interakce prvků

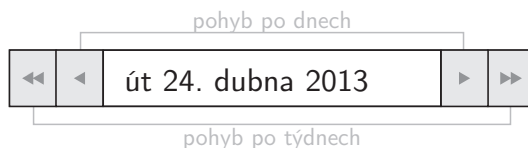
Gadmin nabídne spoustu nových možností především u tabulkových prvků, kde uživatel obvykle vyžaduje pohodlné procházení, řazení a filtrování položek. Právě tyto funkce nově poskytne vyvíjený systém, a zajistí tak obsluhu rychlý přístup k požadovaným datům.

Dostupnost klávesnicí

Ačkoliv předešlý systém nabízel nadstandardní možnosti pro procházení prvků stránky pomocí kláves, nerespektoval základní pravidla, a to aktivaci prvků ve správném pořadí pomocí klávesy Tab. Gadmin jednoznačně definuje, že první pozice obsadí položky menu a poté postupně obdrží tab-index i prvky formulářů.

Nové ovládací prvky

Nabídka formulářových prvků HTML dokumentu v určitých případech nepokrývá požadavky vývojáře nebo uživatele, jedná se například o výběr datového rozmezí. Gadmin proto implementuje element pro pohodlný výběr data (obr. 3.4). Nově také poskytne uživateli při zadávání objednávky kontextové menu, pomocí něhož bude moci obsluha změnit explicitně typ balení či cenu objednávky.



Obrázek 3.4: Návrh prvku pro výběr data

Kontextové nápovědy (hints)

Aby uživatelskému rozhraní a funkcím systému porozuměl bez větších potíží každý uživatel, budou vybrané prvky doplněny o popisek, který se zobrazí, když nad nimi uživatel podrží kurzor. Eliminuje se tak nejednoznačnost některých prvků a nezkušený uživatel bude informován o funkcionalitě bez nutnosti vše vyzkoušet. Až se bude v aplikaci plně orientovat, může zobrazování vypnout v nastavení svého účtu.

3.4 Souhrn požadavků na nový systém

Následující výčet uvádí požadavky kladené na systém Gadmin včetně těch, které nebudou v rámci bakalářské práce zpracovány.

- Přístup více uživatelů s různými rolemi a přístupovými právy,
- správa zákazníků, řidičů, výdejen a u společností možnost připojení dalších kontaktních osob,
- bezpečné vkládání denních a týdenních objednávek; při chybě musí být uživatel informován,
- zavedení flexibilního určování cen – globální cena, cena zákazníka a cena objednávky, zavedení ceny za balení pokrmu,
- vytváření lístků s poznámkami, připojování poznámky k denním objednávkám,
- správa peněžního deníku a pohledávek zákazníků vůči strav. zařízení,
- správa pokrmů a jídelníčků, automatizace vytváření nového jídelního lístku,
- rozšířené možnosti nastavení a personalizace systému,
- registrace narozenin zaměstnanců a státních svátků, zobrazení prostřednictvím kalendáře a upozornění na liště,
- generování přenositelných a tisknutelných výstupů z databáze dle požadavků uživatele,
- perzistence dat objednávek při změně cen a jiných faktorů.

4 Implementace systému Gadmin

Tato kapitola popisuje, jak probíhala volba vhodných technologií a jakým způsobem byl systém následně realizován. Popis nezachází do detailů zdrojových souborů, k tomuto účelu slouží kompletní dokumentace umístěná přímo ve zdrojovém textu ve formě dokumentačních komentářů.

4.1 Srovnání a volba technologií

Ve chvíli, kdy je podoba nového systému podložena odpovídající analýzou staré verze (kapitola 2) a podrobným návrhem funkcionality schváleným zadavatelem (kapitola 3, příloha A), nabízí se zvolit adekvátní technologie a prostředky k její realizaci. U původní aplikace byl tento krok stejně jako analýza zavedených postupů ve stravovací společnosti vypuštěn a o implementaci případných rozšíření či knihoven se rozhodovalo až v okamžiku, kdy se jejich použití jevílo jako nezbytně nutné.

4.1.1 Generování HTML obsahu

Zdrojový text programu používaného v současné chvíli byl skutečně obtížně čitelný. Svůj podíl na jeho nepřehlednosti nesl i způsob, jakým se generoval HTML obsah – požadovaný HTML dokument na výstup posílala funkce `echo`. V jednoduchých případech, např. při generování hypertextového odkazu, by tento postup ještě nebyl tak nevhodný jako při vytváření tabulek či jiných složitých prvků, jež jsou složeny z mnoha HTML tagů. PHP skripty pak tedy obsahovaly rozsáhlé konkatenované řetězce.

Tyto důvody tedy volaly po změně způsobu generování obsahu jednotlivých nabídek. Aby bylo možné aplikaci pohodlně vyvíjet, v ideálním případě využít objektově orientovaného PHP (tedy až ve verzi 5), zajistit její bezproblémový a rychlý chod a koncipovat ji tak, aby mohla být snadno rozšiřitelná, musí vývojář pro její realizaci zvolit framework.

V dnešní době nabízí Internet širokou škálu webových frameworků, které jsou mnohdy dostupné pod open-source licencí. Po prostudování vlastností několika z nich patřily mezi hlavní favority následující:

- Yii (<http://yiiframework.com>),
- Zend (<http://framework.zend.com>),
- Nette (<http://nette.org>).

Všechny zmíněné projekty pracují s MVC (Model-View-Component) modelem, nabízí validaci vstupů formulářů, kešování, prostředky pro autorizaci a auten-

tizaci, rychlý a bezpečný běh aplikace, DRY (Don't Repeat Yourself) principy a podporují AJAX.

Pro vývoj systému **Gadmin** byl nakonec vybrán framework **Nette**. Nepochybně převyšuje zbylé dva projekty českým manuálem a komunitou, kterou si vytvořil díky tomu, že vznikl právě v České republice. Lokalizace dokumentace však nebyla jediným faktorem, který o výběru **Nette** rozhodl. Na rozdíl od ostatních projektů nabízí také šablonovací jazyk **Latte**, který velmi zjednodušuje vytváření šablon dokumentů, čehož lze právě při vývoji systému **Gadmin** velmi dobře využít.

Další součástí, již ocení jistě každý vývojář, představují silné ladicí nástroje. **Nette** pro tyto účely používá tzv. **Laděnk**u, která standardní chybové hlášení PHP přetransformuje do elegantní, snadno čitelné podoby, a navíc jej doplní o úseky zdrojových textů, v nichž chyba vznikla.

Neopomenutelnou výhodou zvoleného frameworku je důraz kladený na zabezpečení. **Nette** automaticky escapuje jakýkoliv výstup do HTML dokumentu, čímž zabráňuje XSS útokům a stejně tak provádí i sanitaci dat odesílaných do databáze prostřednictvím dotazů, a tím pádem chrání před SQL injection (kapitola 2.2.3, další poskytovanou ochranu uvádí zdroj [9]).

4.1.2 Databáze

Volbu databáze ve velké míře omezují parametry hostingu, na němž bude výsledný systém provozován, k dispozici je totiž pouze MySQL databáze. Ačkoliv existují jiné použitelné varianty srovnatelné se zmíněným databázovým systémem, např. SQLite nebo PostgreSQL, využije **Gadmin** pro uchovávání dat MySQL databázi, protože hosting jinou možnost nenabízí.

Důležitou roli dále hraje volba úložiště jednotlivých tabulek. Spousta webových aplikací dnes využívá pro jejich uchování typ **MyISAM**, jenž ale nenabízí podporu cizích klíčů, která představuje jeden z hlavních požadavků na nový systém – zachování referenční integrity. Ta je zodpovědná za to, že nedojde k narušení vztahů mezi daty z tabulek na sobě závislých. Příkladem, kdy je referenční integrita vyžadována, může být situace, při níž se uživatel pokouší z jedné tabulky odstranit zákazníka navzdory tomu, že má tento zákazník v jiné tabulce zaznamenaný objednávky čekající na vyřízení. Kdyby tak došlo skutečně k jeho smazání, nevyřízené objednávky by „osiřely“, protože odkaz na jejich rodičovský záznam, tj. onoho zákazníka, by ztratil platnost [1]. Proto bylo zvoleno úložiště **InnoDB**, které cizí klíče podporuje a současně zajišťuje také dodržení referenční integrity – změny způsobující její porušení automaticky zavrhnou.

Referenční integritu lze alternativně zachovávat i u **MyISAM** úložiště implementací potřebných podmínek přímo v aplikaci. Framework **Nette** ale primárně (bez určitých modifikací) není schopen pracovat s jiným typem úložiště, než je **InnoDB**.

Díky jím poskytnutým cizím klíčům a zachování jistých konvencí pro pojmenování totiž může automaticky tabulky propojovat a ušetřit tak vývojáře konstruování složitých dotazů – ty sestavují metody tříd vrstvy pro komunikaci s databází `Nette\Database` [10].

4.1.3 Dynamika

Dnes je prakticky vyloučeno, aby existovala netriviální webová aplikace, která uživateli nenabízí dynamiku v reálném čase, aplikace, která musí pro jakoukoliv modifikaci stránky znovu načíst kompletní obsah. Do jisté míry si s určitou podporou okamžité interakce mezi uživatelem a prohlížečem poradí skriptovací jazyk JavaScript, jenž umožňuje kontrolovat data ve formuláři, zobrazovat jednoduchá dialogová okna či měnit vlastnosti prvků DOMu. Jeho interpretace se však v prohlížečích mnohdy liší a celkově se jedná o nepříliš uspořádaný jazyk.

Knihovna jQuery

Zmíněné nedostatky JavaScriptu částečně řeší knihovna **jQuery**, která výrazně zjednodušuje procházení DOM struktury, stírá rozdíly mezi různými interpretacemi jazyka jednotlivými prohlížeči a doplňuje jej o spoustu užitečných funkcí, jako jsou animace, snadná manipulace s DOMem a v neposlední řadě také provádění HTTP požadavků technologií AJAX. Především poslední zmíněný bod představuje velký přínos při vývoji systému **Gadmin**, jelikož ten bude pro změnu zobrazování menu používat právě přenos dat na pozadí prohlížeče, tedy zmíněný AJAX.

Knihovna jQuery UI

Kapitola 3.3 uvádí, že nový systém použije ovládací prvky, které HTML standardně neposkytuje. S pomocí **jQuery** by se tvorba takových objektů oproti použití samotného JavaScriptu výrazně usnadnila, nicméně vývojářský tým této knihovny vyvinul rozšíření **jQuery UI** určené právě pro účely implementace nových prvků. Samo o sobě navíc nabízí širokou škálu elementů a rozšířené možnosti pro animování či zvýšení interakce [11].

Plugin dataTables

Protože bude systém využívat napříč celou aplikací zobrazení informací formou tabulky, uživatel pravděpodobně ocení, když mu program umožní takto prezentovaná data řadit, filtrovat a bezesporu také stránkovat. Implementovat takovou funkcionalitu by bylo poměrně složité a především zbytečné, protože existuje spousta pluginů, které transformují standardní HTML tabulku na vysoce interaktivní prvek, jenž umožňuje prohledávat přes vybrané sloupce a stejně tak je i řadit. Plugin **dataTables** nabízí právě tyto možnosti.

Plugin powerTip

Systém Gadmin je koncipován tak, aby s ním mohla pracovat i osoba, která nebyla zaučena a nemá s ním žádné zkušenosti. Ve většině případů se ale nepodaří designerovi uživatelského rozhraní navrhnout aplikaci, v níž má každý prvek na první pohled zřejmý význam. Z toho důvodu bývají objekty uživatelského rozhraní opatřeny kontextovou nápovědou.

Webové prohlížeče ji interně implementují pomocí atributu `title`, jenž může být přiřazen jakémukoliv HTML elementu [12]. Vzhled a zpoždění jejího zobrazení se však napříč prohlížeči odlišují, navíc takto vytvořená kontextová nápověda nemůže obsahovat text formátovaný pomocí HTML, což při objasňování významu složitějšího prvku mnohdy představuje nedostatek.

Plugin `powerTip` nabízí možnost určit, jakým směrem má být kontextová nápověda orientována, po jaké prodlevě se zobrazí a především umožňuje vzhled boxu s textem (libovolně formátovaným pomocí HTML) přizpůsobit pomocí kaskádových stylů.

4.1.4 Uživatelské rozhraní

U webových aplikací neexistuje příliš mnoho prostředků, jak definovat jejich vzhled. Určitou podporu sice nabízí samotné HTML, avšak jeho možnosti jsou značně omezené – umožňuje změnit řez písma, barvu, rozměry obrázku apod. Vesměs tyto vlastnosti fungují jako formátování podobné tomu, se kterým se běžný uživatel setkává u textových editorů. I kdyby si vývojář vystačil s uvedenými možnostmi, musel by formát definovat pro každý prvek dokumentu.

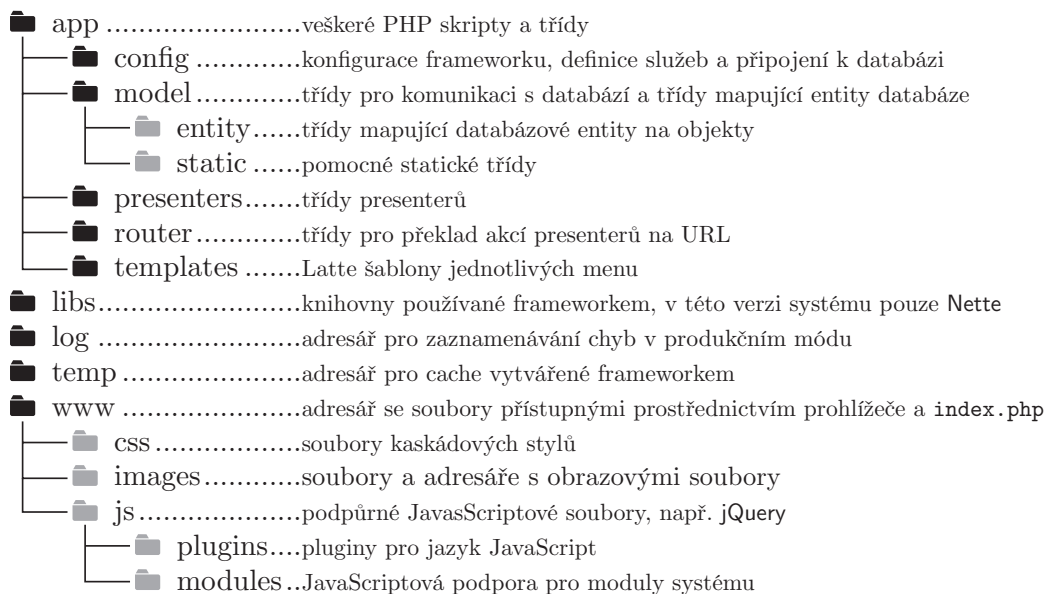
Právě pro takové účely vznikly kaskádové styly (CSS) – jazyk pro popis způsobu zobrazení HTML dokumentů. Nejenže dávají vývojáři stejné možnosti formátování jako HTML, ale rozšiřují je o spoustu dalších, jako jsou např. definice pozicování, rámečků, pozadí a od verze 3 nově také průhlednosti prvků, stínů, animovaných přechodů apod.

Systém Gadmin tedy využije pro stylování uživatelského rozhraní kaskádové styly CSS3 a potřebné obrazové soubory pro realizaci ikon či pozadí ve formátu PNG nebo GIF, jimiž je grafika webových aplikací obvykle doplňována.

4.2 Architektura systému

4.2.1 Adresářová struktura Nette, resp. aplikace

Nette na svých oficiálních stránkách (nette.org) nabízí ke stažení tzv. *sandbox*. Jedná se o předpřipravenou adresářovou strukturu projektu vytvořeného v tomto frameworku. Zároveň tak tato hierarchie složek představuje konečnou podobu organizace zdrojových souborů systému Gadmin (viz obr. 4.1).



Obrázek 4.1: Adresářová struktura frameworku Nette, resp. systému Gadmin (černé složky patří do sandboxu, šedé byly doplněny v rámci aplikace)

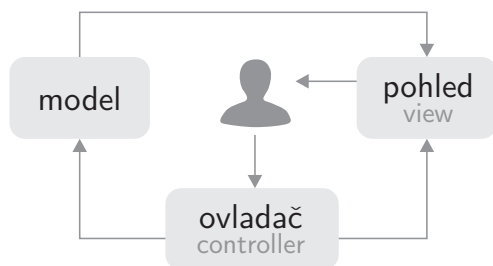
4.2.2 MVC a MVP vzor

Výrazy MVC (Model-View-Controller) i MVP (Model-View-Presenter) označují softwarově-architektonické vzory, jejichž hlavním cílem je oddělit od sebe tři základní vrstvy aplikace s uživatelským rozhraním, a to sice vrstvu aplikační logiky (*model*), zobrazení dat (*view*, *pohled*) a řízení (*controller*, *ovladač*), resp. prezentace dat (*presenter*) [13].

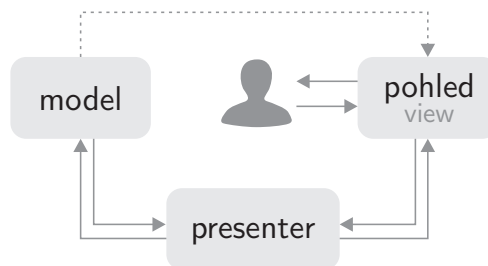
Hlavní rozdíl mezi těmito dvěma pojmy tvoří vztahy jednotlivých vrstev. Zatímco v případě MVC vyvolává uživatel změnu pomocí ovladače a výsledek mu pohled pouze zobrazuje, u MVP vzoru probíhá veškerá interakce výhradně prostřednictvím pohledu [14]. V MVP vzoru tedy plnou kontrolu nad během aplikace získává presenter, který rozhoduje o tom, jaký pohled má být pro zobrazení použit a současně reaguje na akce vyvolané uživatelem, které mu pohled předává. Rozdíly ilustrují obrázky 4.2 a 4.3.

Příkladem funkce MVP může být situace, kdy uživatel v aktuálním pohledu odešle formulář. Pohled vyvolá v presenteru požadovanou akci, v jejímž rámci se provede např. validace formuláře, a následně data odešle do modelu, který je pomocí databázového dotazu uchová.

Ať už se jedná o MVP či MVC vzor, v obou případech je organizace zdrojových textů celého programu mnohem přehlednější a pozdější případné úpravy lze mnohem snadněji realizovat.



Obrázek 4.2: Diagram MVC vzoru



Obrázek 4.3: Diagram MVP vzoru

4.2.3 Detailní model systému Gadmin

Samotný MVP vzor představuje pouze povrchní reprezentaci skutečné struktury systému. Framework **Nette** společně s návrhem **Gadminu** dekomponuje vrstvu modelu na tři samostatné jednotky, viz obrázek 4.4.

Repozitáře

Repozitář funguje jako *služba* pro presenter, poskytuje mu metody k získávání informací z databáze, aniž by sám musel znát názvy tabulek nebo sloupců. V každém cyklu aplikace, tj. při obsluze události vyvolané uživatelem, se vytváří jediná instance potřebných repozitářů, která je přístupná právě presenteru. Ten může se získaným výsledkem manipulovat nebo jej předat zvolenému pohledu, resp. **Latte** šabloně. Služby repozitářů jsou v **Nette** frameworku definovány pomocí konfiguračního souboru (kapitola 4.2.6).

Entity

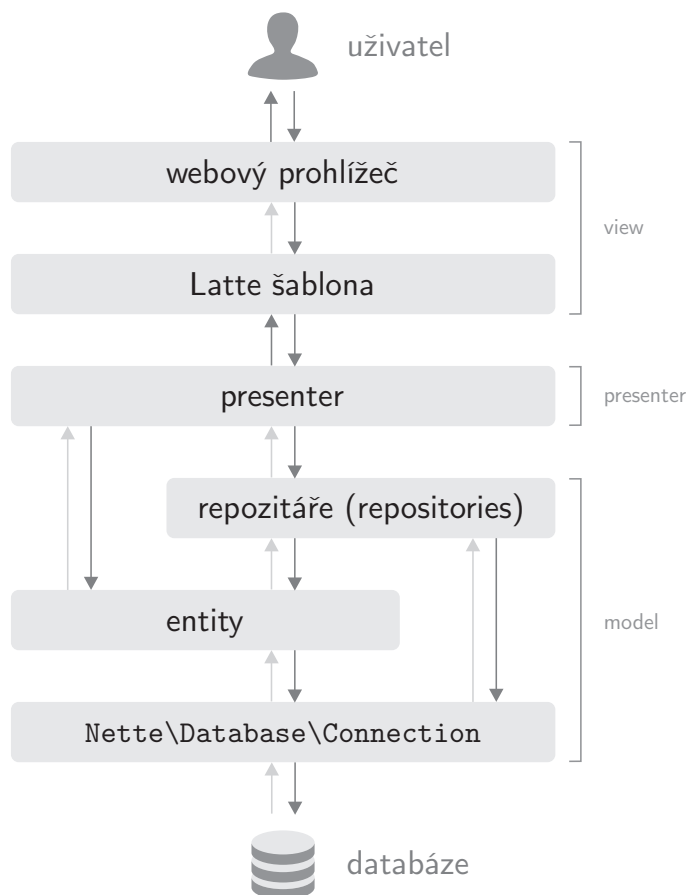
Vrstva entit shlukuje třídy, jejichž instance reprezentují databázové entity, se kterými systém v daném cyklu pracuje. I tyto instance mohou komunikovat s databází, děje se tak například ve chvíli, kdy chce uživatel zjistit, jaký typ balení zákazník vyžaduje. Protože se jedná o konkrétní osobu, nevyžaduje presenter data od repozitáře, nýbrž přímo od instance třídy **Customer** (zákazník). Voláním getteru

```
$packing = $customer->getPacking();
```

tak přímo získá instanci třídy **Packing**. Na pozadí této metody je odeslán dotaz do databáze vyžadující vrácení takového typu balení, jehož ID uvádí instance zákazníka.

Rozdíl mezi entitou a repozitářem lze také chápat na základě vztahů mezi daty. Pokud vývojář požaduje objekty, které jsou nějakým způsobem závislé na jiných (výše zmíněný případ), využije volání jejich getterů.

Naproti tomu ve chvíli, kdy potřebuje obdržet prvky, které nejsou nijak závislé na jiných, má k dispozici repositář (např. výběr všech zákazníků). Z příkladu a z obrázku 4.4 plyne, že přístup k databázi mají jak třídy z vrstvy entit, tak i repositáře a že presenter může pracovat s oběma těmito vrstvami.

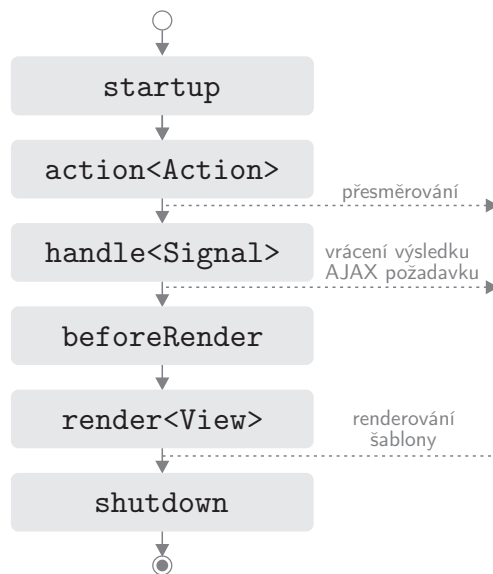


Obrázek 4.4: Detailní diagram struktury systému

4.2.4 Presentery

Předchozí text již několikrát zmiňoval, že presenter slouží především k předávání dat uživateli v lidské podobě. Obstarává prostřednictvím repositářů a entit potřebné informace z databáze, následně je zpracovává a ve výsledku vkládá do pohledu.

Každý presenter při obsluze události vyvolané uživatelem provádí svůj tzv. *životní cyklus*, jehož podobu ilustruje obrázek 4.5. Více informací o presenterech nabízí oficiální dokumentace frameworku [16].



Obrázek 4.5: Životní cyklus presenteru ve frameworku Nette [16]

Životní cyklus presenteru

1. **startup** – inicializuje proměnné a případně pomocí ACL ověřuje, zda může uživatel k presenteru přistupovat.
2. **action<Action>** – existuje pro případ, kdy vývojář chce, aby presenter provedl určitou akci a poté se přesměroval jinam.
3. **handle<Signal>** – zpracovává *signály*, interakce, které nezapříčiňují změnu pohledu, může se jednat např. o AJAX požadavky.
4. **beforeRender** – přizpůsobuje šablony nebo nastavuje proměnné společné pro všechny pohledy (v systému Gadmin jde o zvýraznění aktivního modulu).
5. **render<View>** – připraví pohled, který se má renderovat, a předá mu potřebné proměnné.
6. **shutdown** – ukončuje životní cyklus presenteru.

4.2.5 Pohledy

Pohled předává pomocí **Latte** šablony data získaná z presenteru prohlížeči, jenž je zobrazí uživateli. Z toho plyne, že se musí jednat o HTML dokument, do něhož jsou jen dosazeny vyžadované informace.

Aby **Nette** psaní takových šablon zjednodušilo, implementuje šablonovací jazyk **Latte**. Pokud vývojář uloží standardní HTML dokument jako šablonu (tedy

s příponou `.latte`), může mimo HTML tagy využívat tzv. *maker*, která mu výrazně zjednoduší výpis získaných dat.

Příkladem může být skupina *n-maker*, jež se chovají jako HTML atribut. Hlavní rozdíl ale spočívá v tom, že jejich hodnota při překladu šablony rozhodne o tom, jak bude prvek, v němž se nacházejí, vypadat nebo zda se vůbec ve výstupu objeví. Jedním z těchto maker, které systém Gadmin hojně využívá, je `n:if`. Jeho uplatnění demonstruje následující ukázka.

```
<fieldset n:if='$customer->isCompany()'>
  <div class='legend'>Kontaktní osoby</div>
  ...
</fieldset>
```

Tento úsek kódu vypíše rámeček s kontaktními osobami zákazníka, ovšem jen tehdy, jedná-li o společnost, nikoliv fyzickou osobu, tzn. `isCompany` vrací hodnotu `true`. V opačném případě vypustí celý prvek `fieldset` včetně jeho obsahu.

Jinou ukázkou příkazů Latte představují standardní *makra*. Ta se vždy uzavírají do složených závorek, přičemž bezprostředně za otevírací závorkou musí následovat text příkazu, jinak se složená závorka vyhodnotí jako obyčejný znak. Šablony systému Gadmin velmi často využívají makro `foreach`, jehož použití je následující:

```
<select name='packing'>
  {foreach $packings as $packing}
    <option value='{ $packing->getId()}'>
      { $packing->getTitle() }
    </option>
  {/foreach}
</select>
```

kde `$packings` reprezentuje pole instancí třídy `Packing` (typ balení). Příkaz `foreach` přes něj tedy iteruje a pro každý prvek vygeneruje položku rozevíracího seznamu. Ukázka také zobrazuje způsob výpisu proměnné nebo hodnoty vrácené volanou metodou, a sice `{ $nazev_promenne }` nebo `{ volana_funkce() }`.

Nespornou výhodou Latte při výpisu proměnných jsou tzv. *helpery*, metody, které proměnnou zformátují a teprve poté předají k výpisu. Ačkoliv Nette standardně nabízí poměrně širokou škálu helperů (např. `date` pro formátování data, `truncate` pro zkrácení vypisovaného řetězce a jiné), vývojář může definovat další. Jejich použití vypadá takto:

```
<tr>
  <td>vytvořeno dne:</td>
  <td>{ $datum|date, 'd.m.Y' }</td>
</tr>
```


Příklad očekává, že proměnná `$datum` představuje instanci třídy `DateTime`. Znak svislé čáry pak při překladu upozorňuje, že na ni má být nejdříve aplikován helper, zde konkrétně `date`. Řetězec uvedený za čárkou slouží jako další parametr pro helper, v případě data se jedná o jeho formátování (dle konvencí PHP).

Gadmin implementuje vlastní formátování data `dateLoc`, ceny `price` nebo telefonu `phone` – veškeré dodatečné helpery obsahuje statická třída `Helper`. Další informace o šablonách pohledů, makrech a helperech uvádí oficiální dokumentace [17].

4.2.6 Konfigurace frameworku

Při prvním spuštění sandboxu `Nette` automaticky otestuje, zda běhové prostředí splňuje minimální požadavky (zejména PHP verze 5.2.0 a vyšší). Ale předtím, než je možné s frameworkem `Nette` vůbec začít pracovat, musí vývojář definovat jeho důležitá nastavení. Sám sandbox již obsahuje určitou konfiguraci, s níž je možné jej spustit, taková ale pochopitelně neodpovídá požadavkům systému `Gadmin`.

Soubor `/app/config/config.neon`

Prakticky veškerá konfigurace frameworku se nachází v souboru `config.neon`, jenž dodržuje NEON syntaxi. Ta je svou stavbou velmi blízká formátu YAML, který data zapisuje v podobě čitelné jak pro člověka, tak i pro počítač.

Mimo nastavení časového pásma, expirace záznamů o sezení a výchozího presenteru pro zobrazení chyby, definuje parametry připojení k databázi, které je potřeba před spuštěním aplikace `Gadmin` správně uvést. V případě lokálního serveru mohou vypadat například takto¹:

```
dsn: 'mysql:host=localhost;dbname=gadmin'  
user: 'root'  
password:
```

Další důležitou součástí konfiguračního souboru představuje nastavení a definice služeb, o nichž se text zmiňoval již v souvislosti s presentery. Jako služby registruje konfigurační soubor jednotlivé repozitáře, takže jsou pak dostupné pro presenter. Novou službu lze vytvořit následovně:

```
services:  
  nazevSluzby: nazevRepozitare
```

Kterýkoliv presenter pak může k metodám třídy `nazevRepozitare` přistupovat direktivou `$this->context->nazevSluzby`. Význam repozitářů a služeb přibližovala kapitola 4.2.3.

¹U lokálního serveru není potřeba předávat databázovému spojení heslo.

Soubor /app/router/RouterFactory.php

Třída `RouterFactory` slouží k překladi akcí presenterů na URL dodržující SEO principy pro přizpůsobení stránek internetovým vyhledávačům. Taková optimalizace aplikace ale v případě `Gadminu` nehraje vůbec roli, jednak protože systém není veřejně dostupný a za druhé se jeho veškerý obsah pokaždé načte technologií AJAX, tedy URL nebude viditelná.

Nicméně `RouterFactory` ukrývá jiné, podstatné nastavení, a to sice definici výchozího presenteru, v případě systému tedy menu, které bude zobrazeno při spuštění aplikace (pokud bude uživatel úspěšně autentizován). Následující příkazy uvádí způsob, jakým se standardní presenter ve třídě `RouterFactory` určí.

```
public function createRouter() {
    $router = new RouteList();
    $router[] = new Route('<presenter>/<action>[/<id>]',
        'Board:default');
    return $router;
}
```

První parametr konstruktoru třídy `Route`² určuje, jakým způsobem bude akce presenteru transformována na URL, a druhý parametr pak výchozí akci a presenter, v tomto případě modul `Nástěnka`. Více informací poskytne zdroj [15].

4.2.7 Databáze

Návrh podoby databáze systému `Gadmin` vycházel z diagramu funkčních celků aplikace (příloha A), jenž detailně popisuje případy užití a možnosti jednotlivých modulů. Při sestavování ERA modelu byla obzvláště věnována pozornost uchovávání tras distributorů a jednotlivých objednávek.

Databáze původního systému nesplňovala ani podmínky I. normální formy, protože trasa řidiče nebyla ukládána atomicky [1] (viz kapitola 2.1.5). Nový návrh tedy data maximálně dekomponuje, tak aby byly záznamy atomické a splňovaly III. normální formu. Jeho kompletní podobu³ ilustruje příloha B.

Realizace databáze vyžadovala použití pohledu⁴, `customerNotInRoute`, jenž specifikuje zákazníky, kteří se nenacházejí na žádné z tras. Pohled tvoří sice poměrně jednoduchý dotaz, ale nelze jej realizovat v `Nette` frameworku podobným způsobem jako ostatní. Pohled tvoří následující příkaz:

²`Nette\Application\Routers\Route` je třída frameworku `Nette` a nemá nic společného s třídou `Gadmin\RouteItem`, která reprezentuje položku trasy distributora.

³Příloha B popisuje pouze tu část databáze, která je součástí bakalářské práce. Nezobrazuje tedy tabulky pro moduly `Finance`, `Nastavení` apod.

⁴Nejedná se o pohled, který zmiňovaly kapitoly zabývající se MVC vzorem, nýbrž o databázový objekt podobný tabulce.

```

SELECT customer.* FROM customer
  LEFT JOIN routeItem
    ON routeItem.customer_id = customer.customer_id
 WHERE routeItem.position IS NULL
 ORDER BY CONCAT(customer.surname, customer.name)

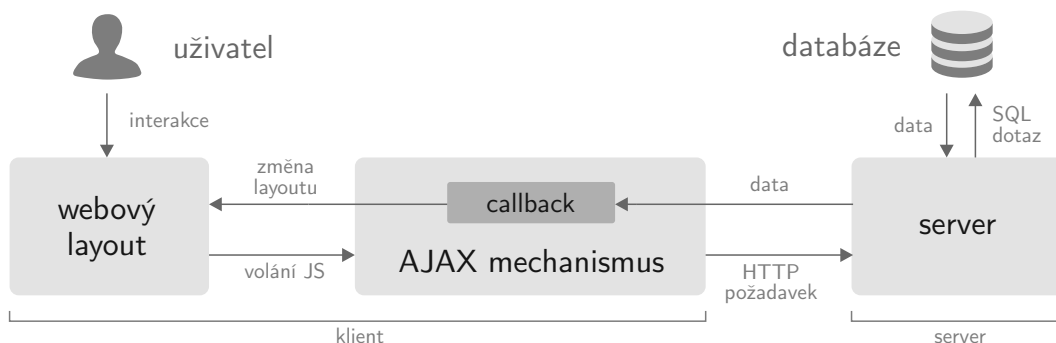
```

Nalezne tedy takové zákazníky, pro něž neexistuje záznam v tabulce `routeItem` (připojený sloupec `routeItem.position` je nulový), a seřadí je dle jména a příjmení. Dotaz úmyslně využívá MySQL funkci `CONCAT`, jež spojuje dva a více řetězců. Sloupec `customer.surname` je u firemních zákazníků totiž nulový, takže zmíněná funkce vrátí pouze hodnotu `customer.name`, čímž se docílí správného abecedního seřazení jak firemních zákazníků, tak i jednotlivců.

4.3 Popis obecné funkcionality

4.3.1 Navigace a AJAX

Moderní webové aplikace načítají téměř veškerý svůj obsah technologií AJAX. Pomocí JavaScriptu vyšlou požadavek na server, ten jej zpracuje (zpravidla získá informace z databáze) a odpověď zašle zpět. Obslužná metoda na straně klienta, tzv. *callback*, obdrží objekt s odpovědí a může na něj adekvátně reagovat např. změnou prvku HTML dokumentu [18], viz obrázek 4.6. Celý proces tedy probíhá na pozadí, aniž by uživatel tušil, že se nějaká data přenášejí.



Obrázek 4.6: Schéma zpracování HTTP požadavku pomocí AJAXu

Důvodů, proč se AJAX tak hojně využívá, je hned několik. Při přechodu uživatele na jinou stránku, resp. modul nebo jeho menu, není nutné načíst celý nový dokument, nýbrž jen jeho vybrané části. Celkově se tak zrychlí práce s aplikací a její chování se přiblíží standardnímu desktopovému softwaru.

S užitím AJAXu lze rovněž plně převzít kontrolu nad pohybem uživatele v systému, nabídnout mu modifikovanou historii a zastavit jej ve chvíli, kdy by

mohl přechodem na jinou stránku ztratit data. Výhoda úpravy seznamu historie spočívá ve vypuštění těch URL, které sloužily pro odeslání formuláře, zamezí se tedy výskytu dialogového okna „Odeslat znovu formulář?“, jež se zobrazuje při návratu na stránku s formulářem.

Gadmin odesílá veškeré takové požadavky s využitím funkce `ajax` implementované knihovnou `jQuery`. Příslušné obslužné funkce, ať už pro oznámení chyby, automatického odhlášení či standardní zpracování odpovědi, poskytuje objekt `Ajax` (`Ajax.js`). Volání funkce `ajax` pro událost kliknutí (`click`) hypertextových odkazů (které nejsou označeny třídou `notAjaxize` – „nezajaxovat“) vypadá následovně:

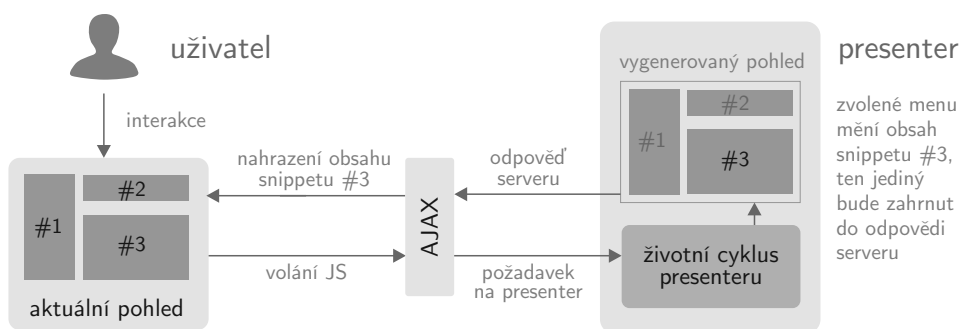
```
function(event) {
    ...
    event.preventDefault();
    ...
    $.ajax(this.href, {
        cache: false,
        dataType: 'json',
        isBack: false,
        isForm: false,
        success: Ajax._success,
        beforeSend: Ajax._beforeSend,
        error: Ajax._error
    });
}
```

kde `event` představuje objekt události klepnutí a identifikátor `this` referenci na odkaz, který akci inicializoval. Volání příkazu `preventDefault` umožňuje zablokovat obvyklé chování při této události, v případě hypertextového odkazu tedy přechod na požadovanou stránku. Z příkladu je patrné, že funkci `ajax` se předávají dva parametry, a sice cílová URL (hodnota atributu `href`) a JSON objekt s konfigurací (detaily uvádí specifikace [19]). Vlastnosti `isBack` a `isForm` nejsou standardní součástí konfigurace, jedná se o přidané atributy definující, jakého typu požadavek je – zda uživatel odesílá formulář nebo se vrací o krok zpět v historii. Pokud jsou oba nepravdivé, zařadí objekt `Ajax` tento požadavek do zásobníku historie procházení.

Atributy `success`, `beforeSend`, `error` určují, jak se má aplikace zachovat při klíčových událostech zpracování AJAX požadavku. Za obzvlášť podstatný lze považovat `success`, jenž aplikace volá v okamžiku, kdy dojde k úspěšnému získání odpovědi od serveru. AJAX mechanismus této funkci jako argument předává objekt, který vrátil server.

Snippets

Použití frameworku **Nette** výrazně zjednodušuje proces *ajaxifikace* celého systému. Nabízí totiž **Latte** makro `{snippet}`, jež umožňuje označit a pojmenovat úsek HTML dokumentu, potažmo **Latte** šablony, nazývaný *snippet* (výstřižek). Pokud je pak proveden požadavek na akci presenteru, jejíž výsledný pohled obsahuje snippets, jejich obsah dosadí presenter do objektu odpovědi serveru. Callback funkce požadavku pak nahradí stávající obsah snippetu tím, jež získá v objektu odpovědi. Proces ilustruje obrázek 4.7. Vývojář může stanovit, které snippets vyžadují změnu svého obsahu, a to tzv. *invalidací*, tedy označením snippetu jako neplatného. Takový presenter pak zahrne do odpovědi odesílané v rámci AJAX požadavku [20].



Obrázek 4.7: Proces výměny obsahu snippetu pomocí AJAXu

4.3.2 Autentizace a autorizace

Významem těchto dvou pojmů a rozdíly mezi nimi se podrobněji zabývá kapitola 3.1.4. Aby mohl být uživatel autorizován pro akce jednotlivých modulů, musí nejdříve projít autentizací. Ta v systému **Gadmin** probíhá ihned při jeho spuštění. Každá osoba oprávněná aplikaci používat disponuje vlastním uživatelským jménem o maximální délce deseti znaků a heslem, jež zadává k ověření své totožnosti.

Prohlížeč může uživatele z bezpečnostních důvodů automaticky odhlásit při dlouhodobé nečinnosti. Protože se tak ale může stát například při zanášení počtu pokrmů do systému, tedy při používání editoru objednávek, který plně využívá AJAXu, nabízí aplikace možnost obnovení přihlášení bez nutnosti opětovného načtení celé stránky. Ve chvíli, kdy vyprší platnost sezení a obsluha programu vyvolá požadavek odeslaný AJAXem, ověří presenter, zda je uživatel přihlášen, a pokud ne, odešle v odpovědi také informaci o tom, že došlo k automatickému odhlášení. Callback funkce na to zareaguje zobrazením dialogového okna, které umožňuje sezení obnovit.

Jakmile uživatel vstoupí do systému, musí se při každé jeho akci ověřit, zda má právo ji provést. Může se jednat o pouhý přesun do jiného menu, ale také o edi-

taci informací apod. Pro tyto účely slouží tzv. ACL, seznam oprávnění přístupu. Pomocí ACL vývojář specifikuje

- uživatelské role, které rozlišují přístupující osoby (např. řidič, distributor, host),
- zdroje, jež označují objekty, u nichž je přístup ověřován (v případě **Nette** frameworku se jedná o presentery),
- a oprávnění, která definují akce zdroje, k nimž uživatele daná role opravňuje.

Gadmin v rámci bakalářské práce pracuje pouze s jedinou uživatelskou rolí (root), jež má oprávnění ke všem akcím. V budoucnu ovšem vzniknou také role **driver** pro umožnění omezeného přístupu řidičům a **guest** sloužící pouze pro prohlížení, nikoliv modifikaci dat. **Nette** nabízí umístění definice ACL do konfiguračního souboru (kapitola 4.2.6). Více informací o sestavování statického ACL v tomto frameworku uvádí článek [21].

4.4 Funkční moduly

4.4.1 Zákazníci

Modul, který představuje základní stavební kámen celého systému. Jedná se o úplnou správu zákazníků, přičemž každý z nich má svoji vlastní kartu s kontakty, detaily odběru a statistikami.

Kontaktní osoby

Speciálním přáním zadavatele byla možnost rozšiřování firemních zákazníků o dodatečné kontakty. Společnost totiž mnohdy uvádí své oficiální, globálně používané telefonní číslo, nicméně objednávky velmi často vyřizuje konkrétní osoba s jiným kontaktem (telefonem, e-mailem).

Každý zákazník typu **společnost** je tedy rozšířen možností **přidat kontaktní osobu**, přičemž jejich celkový přehled nabízí přímo karta zákazníka. Odsud lze také kontakty odebírat.

Odstranění zákazníka

Nový systém na rozdíl od předchozího uchovává veškerá data neomezeně. Pokud tedy existují rok staré objednávky, musí být stále propojeny se zákazníkem, který byl tehdy jejich původcem.

Pokud by **Gadmin** spoléhal pouze na referenční integritu zmiňovanou v kapitole 4.1.2, nemohl by uživatel odstraňovat zákazníky, kteří mají jednu a více objednávek. Ve většině případů ale administrativního pracovníka stravovny nevede nic k tomu, aby uchovával osoby, jež si kdysi objednaly jediný oběd.

Proto je obsluze aplikace sice nabídnuta možnost **odstranit zákazníka**, ale skrývá se pod ní více než hrubé smazání odpovídající položky v databázi. Při tomto požadavku vykonává program následující příkazy:

```
try {
    $res = $customer->getRow()->delete();
    ...
} catch(PDOException $e) {
    ...
    $customer->getRouteItem()->getRow()->delete();
    $this->update($customer);
    ...
}
```

Systém se pokusí záznam skutečně odstranit a v případě, že dotyčný zákazník není nijak vázán na jiná data, se mu to také podaří. V opačné situaci, tj. při narušení referenční integrity (detekováno vznikem výjimky `PDOException`), zůstane položka zachována, pouze se změní hodnota sloupce `is_visible`, jenž informuje o tom, že zákazník má být skryt.

Předdefinované ceny

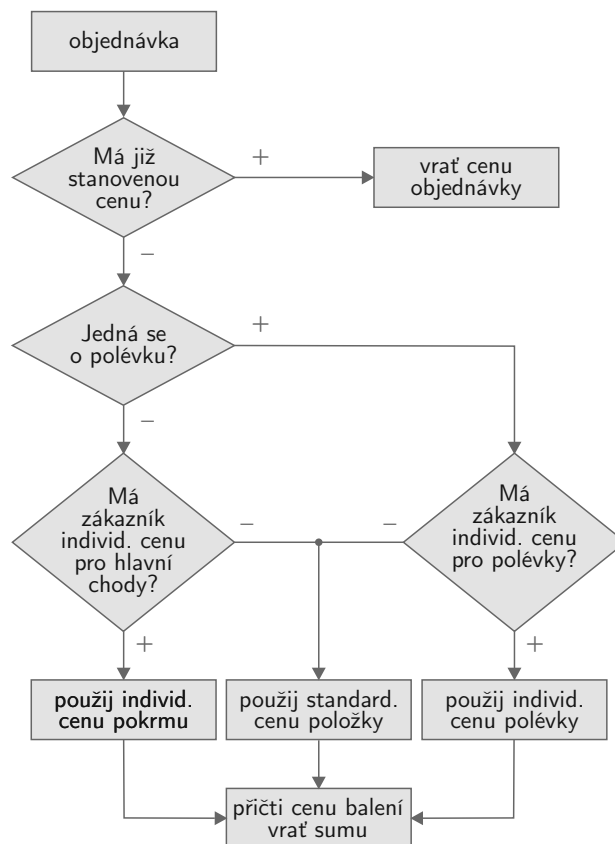
Systém Gadmin pracuje s tříúrovňovým stanovováním cen, o kterém se text zmiňoval již v kapitole 3.2.4, což s sebou přináší oproti původní aplikaci složitější výpočet ceny objednávky. Uživatel může na kartě zákazníka určit, jaká částka mu má být primárně účtována za polévky a za hlavní pokrmy. Způsob, jakým se určuje cena objednávky (kterou může obsluha ve výsledku také změnit), uvádí obrázek 4.8.

4.4.2 Distribuce

Ve stravovacím zařízení existuje několik řidičů a každý z nich obsluhuje svoji vlastní trasu. Mimo řidiče fungují na území okresu také výdejny, které sice nabízejí obědy i náhodným kolemjdoucím, ale stejně tak mohou vést evidenci svých zákazníků. Výdejny a řidiči se tedy v novém systému souhrně označují jako distributoři.

Propojení s uživatelským účtem

Doposud zadávala do systému nové objednávky pouze administrativní pracovníci. Vedení stravovacího zařízení však plánuje přesměrovat tuto činnost přímo na řidiče a to tak, že po dokončení obsluhy trasy a návratu do zařízení, zanesou do systému nové objednávky od „svých“ zákazníků sami.



Obrázek 4.8: Vývojový diagram procesu určování ceny objednávky

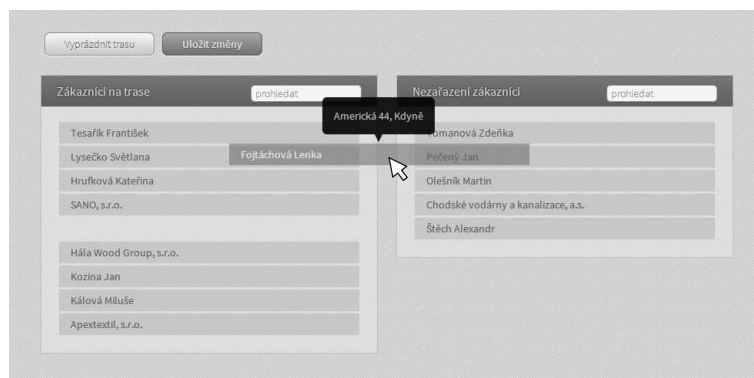
K tomu, aby mohli do systému přistupovat, potřebují přihlašovací údaje, a tedy také uživatelskou roli, pod níž budou v rámci Gadminu vystupovat – driver. Po jejich přihlášení by však nebyl systém schopen určit, která z tras uživateli patří, protože trasa se váže na objekt distributora, který nemá s uživatelem nic společného.

Předchozí odstavec tedy uvádí důvod pro vznik tzv. *spárování* distributora a uživatele. Osoba spojená s řidičem má pak po přihlášení přístup výhradně ke své trase, nemůže tak ovlivnit záznamy ostatních distributorů.

Editace trasy

Systém Gadmin zachovává stejnou koncepci editace trasy jako jeho předchozí verze, avšak snaží se editor položek vylepšit, aby byla manipulace s jednotlivými zákazníky pohodlnější a rychlejší.

Za tím účelem byly oba seznamy (obsluhovaných a nezařazených zákazníků) rozšířeny o prvky pro rychlé prohledávání. Ty umožňují uživateli po zadání dvou a více znaků filtrovat zobrazené položky, čímž redukuje rozsáhlý seznam (mnohdy až se 100 zákazníky) na minimum. Editor ilustruje obrázek 4.9.



Obrázek 4.9: Editor trasy distributora

4.4.3 Objednávky

Modul Objednávky slučuje dohromady dvě služby – vkládání nových objednávek, jejich editaci a současně správu položek přicházejících z e-shopu.

Identifikace odesilatele příchozí objednávky

Aby systém Gadmin usnadnil uživateli práci s příchozími objednávkami, pokouší se automaticky detekovat zákazníka, který objednávku odeslal. Návštěvník e-shopu je povinen při odesílání vyplnit své jméno, telefon a e-mail. Právě poslední dva zmíněné údaje mohou sloužit **téměř** jako jedinečné identifikátory. Nepředpokládá se totiž, že v databázi existuje více zákazníků se společným e-mailem. Pokud tedy aplikace nalezne záznam s odpovídající e-mailovou adresou, navrhne obsluhu jeho spojení s objednávkou, viz obrázek 4.10.



Obrázek 4.10: Ukázka detekce zákazníka

Přijetí příchozí objednávky

Ve chvíli, kdy je objednávka spojena se zákazníkem, může ji uživatel přesunout mezi ostatní schválené objednávky. Před potvrzením lze navíc vybrané příchozí

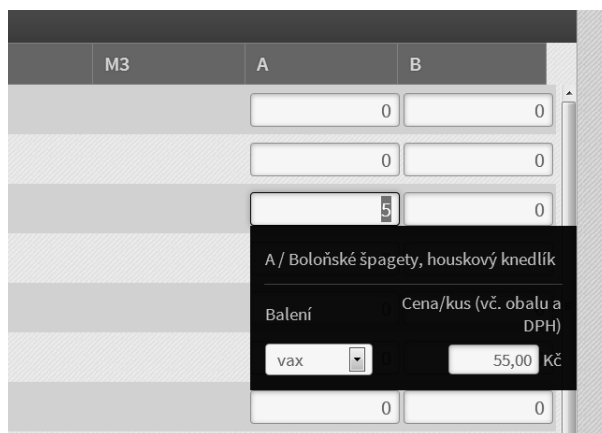
položky odznačit, a zamezit tak jejich přesunu. Funkce přijetí má za cíl minimalizovat faktor lidské chyby při schvalování objednávky. Doposud totiž mohla obsluha přenést příchozí položky pouze ručně, a to přepisem údajů z e-mailové zprávy do editoru objednávek. Nyní stačí klepnutí na jediné tlačítko a aplikace vše provede sama.

Editor objednávek

Jedná se o rozhraní sloužící k hromadné úpravě objednávek v databázi nebo k jejich vložení. V původní verzi systému fungoval čistě jen jako formulář pro zanesení počtu pokrmů, nyní však nově poskytuje rozšířené služby.

Záhlaví tabulky doplňuje kontextová nápověda informující o názvu pokrmu, jenž bude ve vybraný den podáván jako příslušná položka jídelního lístku. Editor také zamezuje zobrazování polí, pro něž na daný den neexistuje záznam v jídelním lístku. Dříve se textové vstupy určující počet pokrmů objevovaly i u pokrmů, které ve skutečnosti ještě neexistovaly – uživatel tedy mohl vytvořit nesmyslnou objednávku.

Gadmin pracuje s tříúrovňovou definicí ceny objednávky, a proto musí umožnit její změnu u každé jednotlivé položky. Navíc na základě požadavku zadavatele poskytuje možnost specifikace balení přímo u objednávky, nikoliv pouze globálně. K těmto účelům je editor vybaven kontextovým menu (obrázek 4.11), jež lze vyvolat stiskem klávesy Ctrl nebo dvojklikem v aktivním poli.



Obrázek 4.11: Ukázka kontextového menu pole objednávky

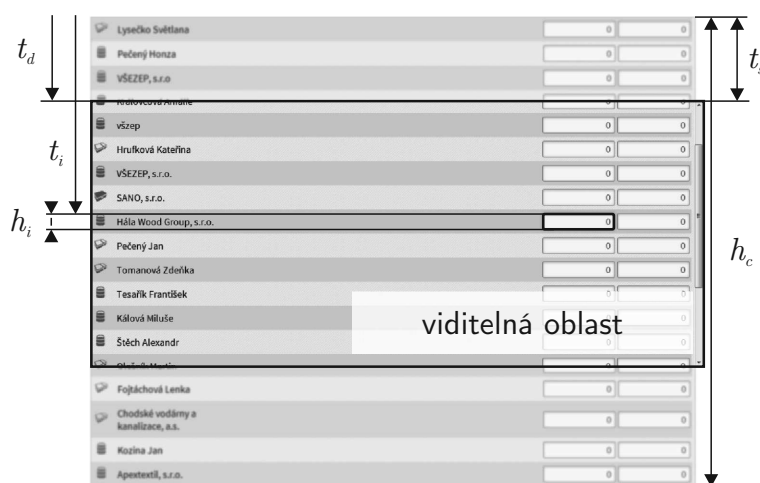
Protože zadávání objednávek představuje poměrně časově náročnou činnost, vyžaduje se, aby během jejího provádění nemohlo dojít ke ztrátě dat. Z toho důvodu ukládá editor objednávek veškeré změny okamžitě, tj. při změně obsahu aktivního pole nebo nastavení v kontextové nabídce – v obou případech tedy ve chvíli, kdy je vyvolána událost `change`. O úspěšnosti ukládání informuje uživatele zbarvení pole do zelena, resp. do červena v případě chyby.

V předchozí verzi systému obsluze programu velmi zneprůjemňovalo zadávání objednávek nepřírozené posouvání obsahu editoru. Pokud uživatel používal pro pohyb mezi textovými vstupy klávesy, očekával, že aplikace sama potřebný obsah editoru odroluje. Ve chvíli, kdy ale bylo aktivní pole skryto pod okrajem viditelné plochy, došlo k jeho zobrazení až po vložení číslice. Nejenže byl tedy aktivní prvek po určité době stále v dolním okraji rolovatelné plochy, ale navíc mohlo také dojít k jeho opožděnému zobrazení.

Tento problém řeší umělé centrování aktivního pole. Při přechodu na jiný textový vstup se vždy odroluje potřebná část obsahu editoru tak, aby nově se aktivovaný prvek zobrazil na středu plochy. Výpočet pozice lze provést dle následujícího vzorce (význam potřebných hodnot ilustruje obrázek 4.12):

$$t'_s = t_i - t_d + t_s - \frac{h_c - h_i}{2}$$

kde t'_s je nová odrolovaná výška, která má být kontejneru editoru nastavena, aby byl prvek vycentrován, rozdíl t_i a t_d představuje vertikální pozici pole vůči rámu kontejneru, po přičtení t_s pak pozici vůči celému obsahu kontejneru. Na vypočtenou pozici se rolování nastaví pomocí jQuery funkce `animate`, která zajistí plynulý posun položek.



Obrázek 4.12: Významy jednotlivých hodnot použitých ve vzorci pro výpočet centrování

4.5 Uživatelské rozhraní

4.5.1 Použité techniky a jejich případné nedostatky

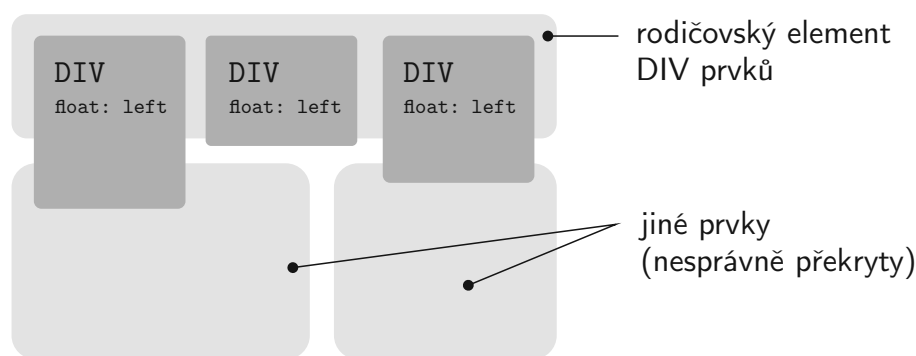
Přetékání DIV elementu

Ačkoliv nová verze kaskádových stylů CSS3 nabízí skutečně revoluční metody pro rozmístění obsahu HTML dokumentu, ne všechny tyto vlastnosti jsou podporovány běžně používanými prohlížeči. Přetrvávají tedy i nadále určité nedostatky starší verzí stylů, jako je například „vytékání“ DIV prvků, jejichž CSS vlastnost `float` nabývá hodnoty `left`. Pokud je takových prvků vedle sebe několik, vyčnívají ze svého rodičovského elementu, jak ilustruje obrázek 4.13.

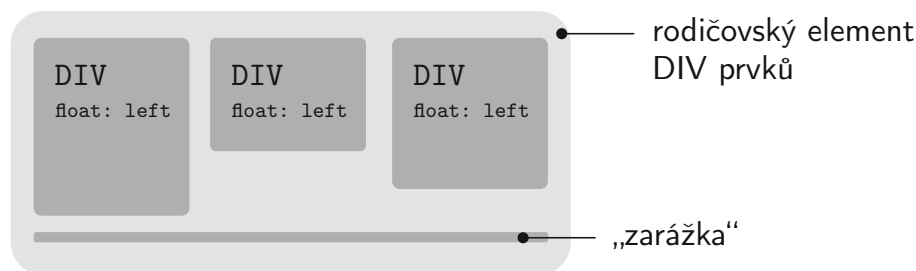
Aby vývojář zamezil takovému chování, musí definovat prvek se speciálním stylem, který bude fungovat jako jakási zarážka. Spodní hranu DIV elementů ohraničí, a donutí tak rodičovský prvek zvětšit svoji výšku na adekvátní hodnotu [22]. Jeho CSS styl vypadá následovně:

```
clear: both;
height: 1px;
margin-top: -1px;
visibility: hidden;
line-height: 1px;
font-size: 1px;
```

Takto ostylovaný prvek nezabírá v dokumentu žádný prostor, protože jeho jednopixelová výška (`height`) je anulována záporným odsazením shora (`margin-top`), a řeší zmíněný problém. Výsledný efekt ilustruje obrázek 4.14.



Obrázek 4.13: Ukázka „vytékání“ prvků z rodičovského elementu



Obrázek 4.14: Výsledný efekt vložení elementu fungujícího jako „zarážka“

Kaskádové styly CSS3

Aby systém Gadmin nesplňoval pouze funkční požadavky, ale byl zároveň také uživatelsky přívětivý a jeho vzhled působil na obsluhu příjemně, používá ke definici vzhledu prvků kaskádové styly CSS3. Seznam všech nových vlastností uvádí zdroj [23]. Systém využívá zejména těchto:

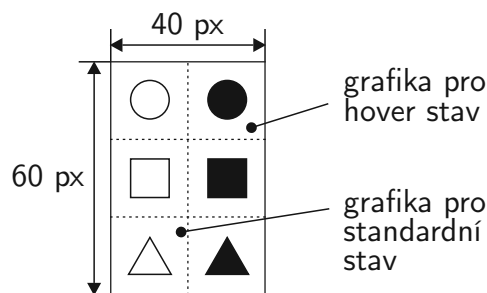
- `border-radius` – zaoblení rohů rámečku,
- `box-shadow` – vnitřní a vnější stín objektu,
- `opacity` – průhlednost,
- `transition` – animace změny CSS vlastnosti.

Většinu z uvedených vlastností ale starší verze prohlížečů přímo nepodporují. Nabízí místo nich atributy s tzv. *vendor-prefixem* [24], předponou specifickou pro každý prohlížeč. Pokud tedy vývojář CSS3 vlastnost doplní o alternativy s vendor-prefixy, zvýší tak pravděpodobnost podpory efektu ve větší škále prohlížečů. Nicméně ani tak si nemůže být jist, že se požadovaný styl zobrazí stejně ve všech verzích.

Soubory ikon a direktiva data

Pokud vývojář vytváří prvky, které disponují tzv. *hover efektem*, změnou podoby ve chvíli, kdy je nad nimi podržen kurzor, a jež jsou tvořeny obrázkem (např. tlačítka s ikonou), musí připravit grafiku pro oba stavy, a obvykle tedy i dva obrazové soubory.

Pokud má uživatel pomalejší internetové připojení, pak při podržení kurzoru nad prvkem zmiňovaného typu dojde k „prokliknutí“, jež je způsobeno prodlevou při stahování obrázku pro hover efekt. Aby tomu vývojář zabránil, může obrázky umístit do jednoho souboru a při aktivaci hover stavu pouze změnit CSS vlastnost `background-position` (viz obrázek 4.15).



Obrázek 4.15: Ukázka organizace obrazového souboru pro tři tlačítka s hover efektem a rozměry 20×20 pixelů

Systém Gadmin u určitých objektů používá velmi malé obrázky. Příkladem může být ikona dárku nebo šipka symbolizující rozbalovací uživatelské menu v informační liště. Aby nemusel pro každý takový prvek existovat samostatný obrazový soubor, umožňuje CSS syntax použít direktivu `data` následovanou base64 formátem dat grafiky [25], například:

```
background: url('data:image/png;base64,iVBORwOKGgoAAAANSUh...');
```

4.5.2 Nové ovládací prvky

Interaktivní tabulka

Aplikace ve velké většině případů zobrazuje data formou tabulky, jejíž obsah mnohdy překračuje hranici sta řádků. Takto přeplněná tabulka začíná být nepřehledná a její procházení se značně znesnadňuje. Proto Gadmin využívá jQuery plugin `dataTables`, který z obyčejného HTML prvku vytváří plně interaktivní element, jenž data automaticky stránkuje, umožňuje je prohledávat a řadit (obrázek 4.16).

Jméno	Adresa	Město	Telefon	Řidič
Apextextil, s.r.o.	Nádražní 679	Kdyně	379 856 999	Ondra
Chodské vodárny a kanalizace, a.s.	Masarykova 388	Domažlice	378 954 645	Ondra
Fojtáčková Lenka	Americká 44	Kdyně	741 547 888	Ondra
Hrufková Kateřina	Mánesova 520	Domažlice	776 123 456	Ondra
Hála Wood Group, s.r.o.	Trhanov 87	Trhanov	897 546 456	Ondra
Kozina Jan	Újezd 1	Újezd	-	Ondra
Kálová Miluše	Mrákov 109	Mrákov	379 778 546	Ondra
Lysečko Světlana	Mánesova 123	Domažlice	602 345 789	Ondra
Olešník Martin	Kout na Šumavě 86	Kout na Šumavě	765 458 987	Ondra
Pešák Jan	Masarykova 457	Domažlice	777 154 987	Ondra

Obrázek 4.16: Tabulka transformovaná na interaktivní prvek pomocí `dataTables`

Podobně jako ostatní jQuery pluginy i tento se inicializuje voláním funkce se stejným názvem, tedy `dataTables`, přičemž se může volitelně předávat také objekt s konfigurací, v případě tohoto pluginu také lokalizací do češtiny. Transformace tabulky může vypadat např. takto:

```
$("#customerTable").dataTable({
  "oLanguage" : Gadmin._DATA_TABLE_LANG,
  "sDom" : '<"dataTables-headbar"<"content"<"filter"f>...>',
  "iDisplayLength": 25,
  "aaSorting": [[ 1, "asc" ]],
  "aoColumns": [
    { "bSortable": false },
    { "asSorting": ["asc", "desc"] },
    { "bSortable": false },
    { "asSorting": ["asc", "desc"] },
    { "bSortable": false },
    { "asSorting": ["asc", "desc"] },
    { "bSortable": false }
  ]
});
```

kde konstanta `Gadmin._DATA_TABLE_LANG` obsahuje objekt s českou lokalizací, vlastnost `sDom` definuje, jak má vypadat DOM struktura tabulky, `aaSorting` uvádí, který sloupec má být standardně řazen, a `aoColumns` způsoby řazení zbylých sloupců. Více informací o jednotlivých vlastnostech konfigurace popisuje oficiální dokumentace [26].

Výběr data

Prvek `dayPicker`, jak je označován i ve zdrojovém textu, byl implementován v rámci vývoje systému a podobně jako ostatní jQuery pluginy i tento využívá rozhraní pro tvorbu rozšíření `Widget` z knihovny jQuery UI. Zároveň staví na již existujícím pluginu pro podobné účely, a sice na `datePickeru`, který ovšem zobrazuje pouze kalendář.

Za hlavní přednost `dayPickeru` lze považovat především přidaná tlačítka pro rychlý posun. Ta umožňují uživateli měnit vybrané datum po dnech či týdnech, a to bez nutnosti hledání v kalendáři. Plugin navíc ignoruje víkendy, obsluha tedy nemůže zvolit datum mimo pracovní dny, `dayPicker` automaticky po pátku zobrazuje pondělí.



Obrázek 4.17: jQuery plugin dayPicker

4.5.3 Prostředky nápovědy

Jeden z hlavních aspektů, na který byl při vývoji systému Gadmin obzvláště kladen důraz, představovala snaha o docílení vysoké míry intuitivnosti a jednoduchosti ovládání i pro nezaškolené uživatele. Aplikace, jež aspiruje na takové vlastnosti, nesmí postrádat efektivní nápovědu. Ta by měla být uživateli dostupná v kterémkoliv okamžiku a ve stručné a srozumitelné formě.

Kontextová nápověda

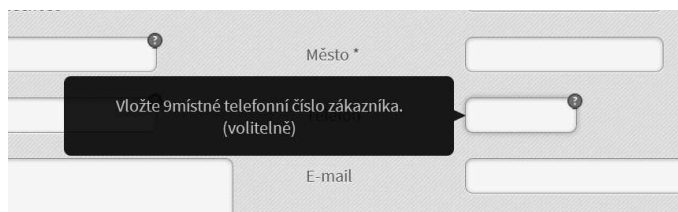
Tou nejstručnější formou, kterou systém uživatele informuje o funkcionalitě prvků, je kontextová nápověda. Existují prvky, jejichž význam nemusí být na první pohled nebo při prvním použití okamžitě jasný. Podrží-li uživatel u takových prvků kurzor, zobrazí se „bublina“ se stručným popisem objektu.

Aplikace pro tento účel používá jQuery plugin `powerTip`, jenž umožňuje určit pozici boxu s nápovědou vůči prvku (navíc disponuje mechanismem, který je schopen automaticky detekovat ideální umístění) a především pak použít HTML formátování. Toto rozšíření vyžaduje, aby prvky, u nichž se má pomocný popis zobrazit, obsahovaly tzv. *data-atribut*, konkrétně `data-powertip`, jehož hodnotu představuje právě text nápovědy. Při načtení stránky pak musí být volána konstrukce, která ke všem takových elementům na základě jejich *data-atributu* vytvoří prvek „bubliny“. Ta vypadá následovně:

```
$("#*[data-powertip]").powerTip({
  smartPlacement: true,
  fadeInTime: 300;
});
```

Volání jQuery funkce (\$) s tímto parametrem vybere z DOMu dokumentu všechny prvky s atributem `data-powertip` a aplikuje na ně plugin `powerTip`.

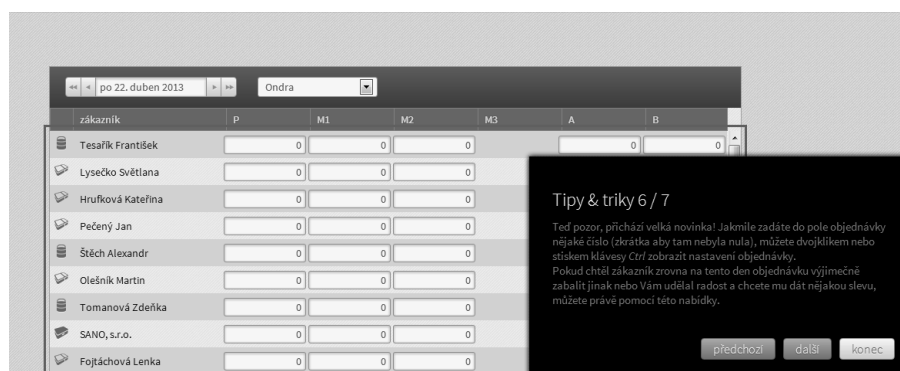
Pokud se jedná o prvek formuláře, označí jej Gadmin navíc ikonou otazníku, čímž uživatele upozorní na to, že má pro toto pole k dispozici doprovodný text, viz obrázek 4.18.



Obrázek 4.18: Kontextová nápověda pro pole formuláře

Průvodce Tipy & triky

Uživateli, který se se systémem Gadmin nikdy nesetkal, by samotné kontextové nápovědy patrně nepomohly. Proto je aplikace vybavena průvodcem **tipy&triky**. Téměř každé menu obsahuje vlastní sadu stručných textů, jež obsluhu programu rychle seznamují s funkcemi, které by mohly ocenit nebo o jejichž existenci by se za normálních okolností dozvěděli jen zřídka. Uživatel tak není nucen číst rozsáhlé příručky, z nichž si stejně nebude vše pamatovat, protože může kdykoliv během práce se systémem potřebné informace získat právě z průvodce (obrázek 4.19).



Obrázek 4.19: Interaktivní průvodce aplikací

Tipy&triky se zobrazují po klepnutí na odpovídající tlačítko v menu aktivního modulu. V pravém dolním rohu obrazovky se po jejich vyvolání objeví černý rám s textem prvního tipu. Zároveň systém označí červeným rámečkem prvek, k němuž se nápověda vztahuje. Uživatel má k dispozici tlačítka, pomocí nichž

může mezi tipy a triky přepínat. **Gadmin** při každém přechodu odroluje obsah stránky tak, aby byl textem zmiňovaný prvek dobře viditelný.

Definice tipů a triků je realizována pomocí **Latte** makra `{block}`, jež umožňuje dosazovat část z jedné šablony do jiné. Hlavní layout aplikace tedy očekává, že šablona modulu obsahuje blok s tipy a triky, a pokud jej skutečně nalezne, dosadí ho do výsledného dokumentu a jednotlivé položky pak po zobrazení průvodce prochází pomocí **jQuery**. K tomu je ale také zapotřebí, aby presenter předal šabloně proměnnou `$showHints`, která zajistí zobrazení tlačítka pro vyvolání průvodce.

5 Testování

Jedním z hlavních úkolů této bakalářské práce bylo provést adekvátní testování vyvinuté aplikace, resp. její implementované části. Následující kapitoly se zabývají způsoby, jež mohou být pro testování použity, uvádí, které z nich se pro ověření funkčnosti a použitelnosti systému hodí, a vyhodnocují získané výsledky.

5.1 Dostupné metody testování

Protože u webových aplikací existuje spousta aspektů, jež je možné prověřovat, lze metody testování kategorizovat. Následující výčet uvádí vlastnosti systému, které je možné testovat.

- Kompatibilita s prohlížeči,
- bezpečnost,
- zatížení aplikace,
- požadovaná funkcionalita,
- použitelnost a intuitivita rozhraní.

Většinu těchto aspektů lze testovat pomocí aplikací vyvinutých k tomuto účelu. Mezi velice populární patří např. nástroj **Selenium**, jenž poskytuje vývojářům možnost definice automatických testů. Protože ale bude aplikace testována reálnými osobami, není potřeba služby tohoto nástroje využívat.

Systém **Gadmin** je postaven na frameworku **Nette**, u něhož jsou rizika bezpečnostních mezer a případných útoků již ošetřena [9], proto z hlediska bezpečnosti nebude také nijak testován. Totéž platí pro zatížení aplikace – vzhledem k tomu, že program mohou používat pouze zaměstnanci stravovacího zařízení, nehrozí nebezpečí toho, že by mohla zkolabovat vlivem přetížení. Ostatní aspekty testování řeší následující kapitoly.

5.2 Testování kompatibility

Systém **Gadmin** je primárně doporučeno spouštět v prohlížeči **Google Chrome**. Tuto skutečnost zadavatel přijal, a proto nebyl brán zřetel na interpretaci systému v jiných prohlížečích. Protože se však jedná o aplikaci splňující validační standardy, i ostatní prohlížeče by měly být schopné umožnit (aspoň částečně) korektní chod **Gadminu**.

Pro ověření kompatibility byli zvoleni přední zástupci v současnosti používaných programů pro prohlížení webových stránek [27]. Navíc je doplnil i **Internet Explorer** verze 7, který doposud stravovací zařízení používalo. Následně vznikl

seznam potenciálně problémových oblastí systému, jež by mohly v prohlížečích způsobit chybu či fungovat nesprávně. Ve většině případů se jedná o funkce, které pracují s novými vlastnostmi kaskádových stylů či JavaScriptem. Testované oblasti a jejich podporu ze strany prohlížečů popisuje příloha D.

Výsledek testování kompatibility

Příloha zcela jasně potvrzuje plnou kompatibilitu prohlížeče Google Chrome se systémem Gadmin. Současně lze ale i další dva konkurenční programy, Opera a Mozilla Firefox, považovat za aplikace, jež práci se systémem v omezené míře umožňují. Nutno zdůraznit, že v případě Opery však musí uživatel dbát na to, aby nepoužíval navigaci prohlížeče, tu zde totiž není možné blokovat jako u ostatních, a může tak dojít ke ztrátě neuložených dat.

Nejhůře v testech dopadly prohlížeče Internet Explorer, které nepodporují prověřované vlastnosti Gadminu ani ze 75 %, a to v obou zmiňovaných verzích. Ačkoliv lze říci, že verze 9 může být v krajních případech pro práci se systémem používána (protože její nedostatečná podpora kaskádových stylů omezuje pouze vzhled, nikoliv funkčnost), rozhodně to nemůžeme tvrdit o verzi 7, která aplikaci takřka vůbec nepodporuje.

5.3 Manuální testování

Hlavní roli při ověřování funkčnosti a celkové použitelnosti systému hrály manuální testy, tedy testy prováděné skutečnými osobami. Jejich cíl představovalo jak prozkoušení požadované funkcionality, tak i posouzení přívětivosti a intuitivity uživatelského rozhraní. Pro tyto účely vznikl testovací arch (příloha C), jež se skládá ze tří částí:

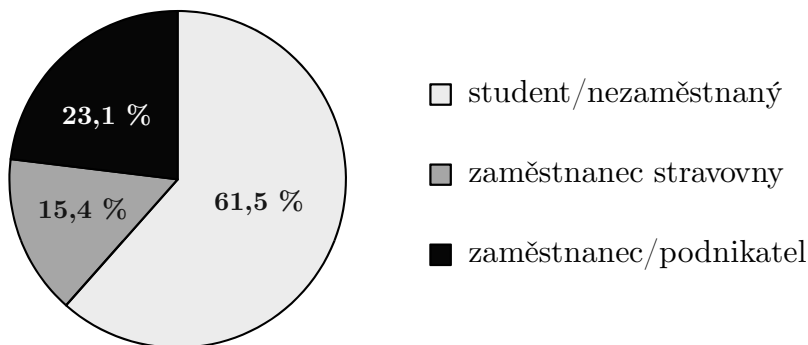
- profil testera, který umožní pracovat s výsledky relevantněji,
- testovací scénář, jež prověří funkcionality implementovanou v rámci bakalářské práce a zároveň také orientaci uživatele v rozhraní aplikace,
- a závěrečný průzkum sloužící k písemnému vyjádření testerů, k vytvoření prostoru pro sdělení připomínek, návrhů apod.

Pro testování byl zvolen vzorek dvanácti testerů tak, aby se jednalo o osoby s různými zájmy, profesemi, popř. obory, v nichž se vzdělávají. Graf na obrázku 5.1 uvádí složení osob na základě jejich profesního stavu a na obrázku 5.2 pak procentuální zastoupení dovedností mezi všemi testery.

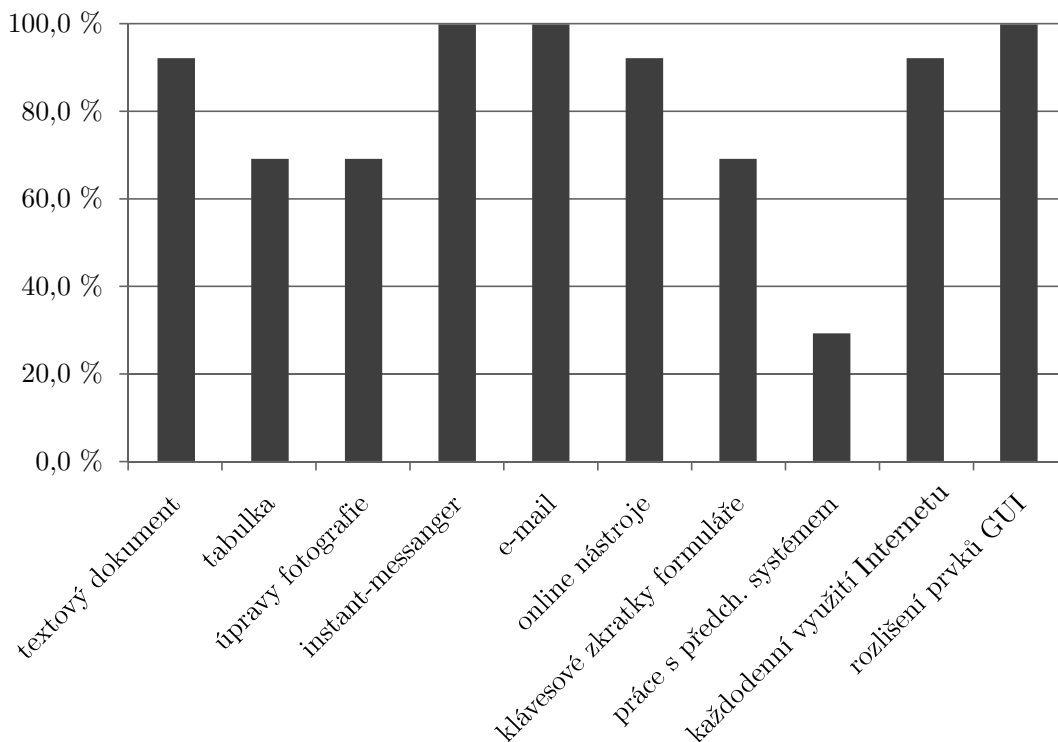
Co se týče samotného průběhu testování, měl uživatel k dispozici nejdříve tzv. *hřiště*, instalaci aplikace speciálně určenou k experimentům, kde mohl tester dle své libosti manipulovat s daty, a tím si tak práci se systémem osvojit.

V druhé fázi pak přešel k jiné instalaci systému, jejíž součástí je databáze předpřipravená pro testovací scénář. S přechodem do této fáze se také inicializoval

skript **TestLogger**, jehož hlavní náplň spočívá v zaznamenávání veškeré testerovy interakce, především pak přechodů mezi moduly či menu (požadavky AJAX). Všechny tyto záznamy **TestLogger** vkládá do jediného souboru, jenž tester na konci plnění scénáře musí uložit. Dalším zpracováním tohoto souboru pak bylo možné odhalit, která z testových částí uživateli trvala nejdéle, zda ji skutečně splnil nebo jestli požadovanou funkci hledal v jiném menu.



Obrázek 5.1: Graf znázorňující složení testerů na základě jejich profesního stavu



Obrázek 5.2: Graf procentuálního zastoupení dovedností tak, jak je uvedli testeři

5.3.1 Výsledky testovacích scénářů

Průběh testování u jednotlivých uživatelů tak, jak jej zaznamenal **TestLogger**, uvádí graf v příloze E. Zobrazuje časy, v nichž uživatel dokončil danou úlohu v rámci testovacího scénáře. Křivka testera tedy určuje celkovou délku testování, ale také časovou prodlevu mezi jednotlivými kroky. Již při prvním pohledu na tento graf je patrné, že největší potíže pro testery představoval krok, při němž měli vložit novou objednávku do databáze. U tohoto úkolu totiž ve většině případů křivka průběhu strmě roste, což signalizuje větší časovou prodlevu. Celkovou náročnost celé testovací části III navíc dokresluje také graf v příloze F, jenž ji označuje v porovnání s ostatními za nejobtížnější (s průměrem 2,25).

Podobný nárůst je sice patrný i u vkládání nového zákazníka a distributora, avšak při těchto dvou krocích musí tester do formuláře zanechat textové údaje. Pokud tedy neovládá psaní všemi deseti prsty či je jeho schopnost psaní na klávesnici jinak omezena, zvětší se prodleva i u těchto dvou úkonů. Někteří z uživatelů, např. tester A, F či H, vybrané úkoly neprovedli správně, nebo dokonce vůbec, proto je jejich křivka přerušena. Za ideálního uživatele aplikace lze považovat testera G, jehož průběh práce s **Gadminem** je zcela plynulý, bez chyb a ve výborném čase 14 minut a 50 sekund.

V rámci manuálního testování byl celý scénář u třetiny uživatelů proveden znovu s odstupem 2 dnů. Toto opětovné testování si kladlo za cíl odhalit, zda po prvotním seznámení se systémem bude uživatel schopen pracovat s jistou znalostí aplikace rychleji. Jak ukazuje příloha G, graf rozdílů prodlev mezi úkoly v prvním a druhém testování, ve většině případů došlo v průměru k minutovému zlepšení. Obzvláště patrná změna se pak projevila v oblasti, která byla v první fázi testování vyhodnocena jako kritická, tedy v úkolu vkládání objednávky.

5.3.2 Výsledky závěrečného průzkumu

Protože se jedná o písemnou část testovacího archu, nelze její výsledky sjednotit ani nijak globalizovat. Z toho důvodu se budou jednotlivým otázkám průzkumu věnovat následující podkapitoly.

Vzhled a prvky uživatelské rozhraní

Všichni uživatelé, kteří se testování účastnili, považují vzhled systému **Gadmin** za moderní a přehledný, líbí se jim. Ve většině případů souhlasí i s volbou barevných kombinací, pouze tester C uvádí, že v určitých případech shledává barvy písma a pozadí málo kontrastními, má obtíže se čtením těchto úseků.

Několik uživatelů dále poznamenalo, že velikost navigačního tlačítka **Zpět** je v porovnání s ostatními prvky menu velmi malá, musí tedy kurzor myši umisťovat přímo na ikonu šipky.

Tester C navíc uvedl, že považuje mezeru oddělující menu modulu a jeho samotný obsah za zbytečnou. Obecně nabyt během testování dojmu, že ve spoustě

případů musel příliš mnoho rolovat obsah obrazovky. Z těchto důvodů by mezery (tedy CSS vlastnosti `padding` a `margin`) zmenšil či zcela odstranil.

Koncepce menu

Organizaci menu a dalších nabídek shledávali testeři vyhovujícími. Výjimku tvoří umístění editoru trasy distributora. Ten je dostupný pouze prostřednictvím ikony mapového terčíku v tabulce distributorů. Většina uživatelů (66 %) však očekávala (a vyplývá to také ze záznamů skriptu `TestLogger`), že tuto funkci naleznou na kartě distributora, a proto uvedla tuto skutečnost jako nevyhovující.

Menšímu počtu testerů (33 %) se pak jako nedostatek jeví označení editoru objednávek. Ten je momentálně dostupný v modulu **Objednávky** pod položkou **spravovat**. Někteří uživatelé si však nespojili správu objednávek s možností jejich vložení a marně se pokoušeli nalézt prvek s popiskem **vložit**. K této dezorientaci přispěla také vysoká pravděpodobnost záměny příchozí (menu **příchozí**) a přijaté objednávky (menu **spravovat**).

V závěru tester L uvedl, že postrádá zvýraznění podmenu (menu modulu) podobným způsobem, jakým je realizováno v nabídce modulů.

Průvodce tipy&triky

Téměř všichni uživatelé označili tohoto průvodce aplikací za užitečného a v průzkumu zmínili, že díky němu pro ně byla obsluha aplikace snazší (dle záznamů skriptu `TestLogger` použil průvodce každý tester v průměru čtyřikrát). Výjimku tvoří tester C, jenž považuje texty této nápovědy za příliš rozsáhlé. Naproti tomu tester G navrhoval rozšíření průvodce o další užitečné tipy. Celkově lze tedy **tipy&triky** označit za užitečnou formu nápovědy, což možná ilustruje i následující odpověď testera I na otázku, zda je průvodce užitečný:

Určitě více než nápověda ve Windows. I když srovnáváním s ní bych tyto tipy&triky jen urážel.

Hodnocení funkcí

Testeři byli v rámci průzkumu požádáni, aby označili funkce systému **Gadmin**, o nichž si myslí, že jsou užitečné, či naopak. Tři testeři shodně ocenili editor tras a možnost filtrace položek v něm obsažených. Jiní tři označili za užitečnou funkci průvodce, jemuž se věnovala předchozí podkapitola. Tester I považuje za prospěšné okamžité přijetí příchozí objednávky bez nutnosti přepisování počtu pokrmů a tester G uvedl, že se mu líbí koncepce **Nástěnky**, která sice nebyla v rámci bakalářské práce implementována, ale ze systému lze vysledovat, k jakým účelům poslouží.

Tester C se zaměřil především na negativní postřehy a v rámci průzkumu znamenal, že mu nevyhovuje způsob výběru distributora v editoru objednávek.

Výběr je nyní realizován pomocí prvku `SELECT`, tento tester však navrhoval hromadné zobrazení distributorů např. v jedné řádce.

5.4 Výsledky a jejich význam pro další vývoj

5.4.1 Uživatelské rozhraní, intuitivita

Jak uvádí předchozí kapitola, všichni testéři byli se vzhledem rozhraní systému spokojeni a označili jej jako moderní. Několik z nich však považuje tlačítko `Zpět` za příliš malé. `Zpět` je skutečně menší než ostatní položky menu, obsazuje plochu o rozměrech 9×13 pixelů, což lze u jednoho z hlavních navigačních prvků považovat skutečně za nedostatek. Uživatel musí (v závislosti na nastavení myši) pro umístění kurzoru nad tento objekt vynaložit určité úsilí, což ubírá na rychlosti jeho použití. Při dalším vývoji systému bude tedy toto tlačítko rozšířeno na 50 pixelů šířky a 40 pixelů výšky (stejně jako ostatní položky menu).

Nezanedbatelný počet testerů také upozornil na nevhodné umístění editoru tras. Při dalších úpravách systému tedy bude přesunut na kartu distributora, aby byly veškeré informace s ním spojené dostupné v jednom menu.

Tester C ve svém vyjádření uváděl, že považuje za zbytečné mezery mezi jednotlivými prvky menu, že je tak nucen rolovat zbytečně velké množství obsahu. Tyto mezery, tedy vlastnosti `margin` (vnější odsazení) a `padding` (vnitřní odsazení), však mají v layoutu systému svůj význam. Prvky od sebe opticky oddělují a design aplikace celkově „provzdušňují“. Nelze popřít, že díky nim musí uživatel odrolovat více obsahu, nemusí však posouvat obsah stránky zpět – za tím účelem je `Gadmin` doplněn funkcí `nahoru`.

Tester L se trefně zmínil o absenci zvýraznění aktivního podmenu. Stejně jako moduly prostřednictvím ikon indikují, který z nich uživatel aktivoval, i jejich podmenu by mělo na první pohled informovat, kde se obsluha programu nachází. Presentery jednotlivých modulů tedy budou nově mimo proměnnou `$selectedMenu` odesílat šabloně také proměnnou `$selectedSubmenu`, která umožní aktivní podmenu označit.

5.4.2 Funkcionalita

Hlavním nedostatkem v této oblasti se stalo pravděpodobně vkládání objednávky a editace jejího typu balení či ceny – vyplývá to jak ze závěrečného průzkumu, tak i ze záznamů skriptu `TestLogger`, z něhož je patrné, že nejvíce času testerovi nezabralo samotné vložení objednávky, nýbrž nalezení menu, kde může tento úkon provést. Proto dojde v systému k přejmenování menu `spravovat` na `vložit / upravit`, což sice vyjadřuje totéž, avšak se zvýrazněním možnosti vkládání.

Co se týče úprav vlastností objednávky, doposud musel uživatel stisknout v aktivním poli s objednávkou klávesu `Ctrl` nebo v něm poklepat levým tlačítkem myši. Ačkoliv se tento způsob jevil jako snadno odhalitelný zejména proto, že

editace dvojklikem představuje zavedený princip, spousta testerů se o způsobu vyvolání této funkce dozvěděla až s použitím průvodce *tips&tricks* a nebo vůbec. Z toho důvodu bude aktivní pole editoru vždy opatřeno tlačítkem s ikonou šipky (podobným tomu, které využívá prvek `SELECT`), jež uživatele bez pochyb upozorní na skryté kontextové menu sloužící k editaci vlastností objednávky.

Tester C uvedl, že mu nevyhovuje způsob výběru distributora v editoru objednávek, navrholo umístit všechny distributory vedle sebe do jedné řády. Rozdíl mezi návrhem testera a současnou realizací výběru však spočívá v jediném klepnutí myši navíc, jež jen stěží může rychlost práce závratně ovlivnit. Pro zachování současného stavu navíc hovoří skutečnost, že nelze předpokládat žádný předem daný počet distributorů, snadno by pak tedy mohlo dojít k situaci, kdy by se všichni distributoři do záhlaví tabulky editoru nevešli, a zvětšili by tak rozměry záhlaví, nebo naopak nebyli vůbec viditelní.

5.4.3 Prostředky nápovědy

Jak ukázaly výsledky testů, většina uživatelů velmi ocenila průvodce nabízejícího *tips&tricks*. Tester C ve svém archu nevyjádřil, zda *tips&tricks* používal (záznamy z jeho testování uvádí, že nikoliv, avšak průvodce mohl zobrazit také na *hráči*), ale texty považoval za příliš dlouhé.

Při sestavování *tips&tricks* byl kladen důraz na popularizaci příručkových textů – jde tedy o způsob popisu funkcí, který by uživatele neměl nudit a přirozenou formou mu osvětlil význam vybraných prvků uživatelského rozhraní. Lze očekávat, že texty takového charakteru nabudou u vybraných menu větších rozměrů, v případě *Gadminu* se jedná především o správu objednávek, jež vyžaduje rozsáhlejší popis funkcionality. Proto nedojde v budoucnu k jejich zkrácení, naopak bude průvodce rozšířen o další tipy vztahující se jak k novým funkcím, tak i k těm, u nichž testy prokázaly určité nedostatky.

5.4.4 Další úpravy

Testování aplikace *Gadmin* odhalilo i nedostatky, které uživatelé nezpozorovali. Jedním z nich je např. vznik duplicit při nesprávném zacházení s příchozími objednávkami.

Pokud se zákazník nachází v databázi systému, ale přesto při odesílání elektronické objednávky nepoužil své ID, *Gadmin* jej identifikuje, avšak vyžaduje po obsluze, aby jeho spojení s objednávkou potvrdila (případ Jana Pečeného v testovacím scénáři, příloha C). Aplikace na skutečnost, že zákazníka rozpoznala, upozorní zobrazením odkazu s textem „Není to ...?“. Několik testerů to však přehlédlo a zákazníka vložilo do systému, čímž vznikla duplicita lišící se např. jen jménem nebo mezerou nechtěně vloženou do adresy. Systém zákazníky identifikuje na základě e-mailové adresy a telefonního čísla. Pokud by se tedy v budoucnu pokoušel tester provést zmíněné operace, aplikace mu to nedovolí, protože bude

vyžadovat pro každého zákazníka v databázi unikátní e-mail a telefonní číslo. Tyto dva údaje se u osoby objedávající obědy totiž nikdy nemohou opakovat.

Tester C ve svém archu mimo jiné uvedl, že by bylo vhodné, aby se v editoru objednávek pole s položkami, jejichž vlastnosti (typ balení, cena) byly pomocí kontextového menu změněny, nějakým způsobem označovala. Administrativní pracovník zařízení jistě ocení, když v editoru na první pohled rozpozná objednávky, jejichž typ balení či cena se liší oproti standardnímu nastavení u daného zákazníka. Zmíněnou funkcionalitu lze realizovat pomocí odpovídající CSS třídy, o jejímž přiřazení k prvku rozhodne porovnání atributů objednávky s vlastnostmi definovanými u zákazníka.

6 Závěr

Tato bakalářská práce si kladla za cíl vytvořit nový, snadno rozšiřitelný systém správy pro stravovací zařízení. Ve společnosti zadávající tento projekt již podobná aplikace existuje. Ta však nespĺňuje všechny požadavky administrativní pracovnice, a proto byla prvním krokem práce důkladná analýza programu, jež má zmíněná firma momentálně k dispozici.

Na základě odhalených nedostatků mohl být sestaven hrubý návrh nového systému a databáze, který po několika konzultacích následně doplnily další požadavky zadavatele. Při vytváření kompletní podoby návrhu byl obzvláště kladen důraz na logické rozmístění jednotlivých nabídek a jejich smysluplné provázání. Obdobná pravidla pak platila také při návrhu uživatelského rozhraní, které se snaží být maximálně přívětivé a intuitivní. Pro jeho realizaci byly zvoleny moderní kombinace barev, ikony jednoznačně nesoucí význam a především pak prostředky kontextové nápovědy, jež mají uživatele ušetřit listování v příručce.

Protože měl být nový systém zvaný *Gadmin* snadno rozšiřitelný, byl k jeho implementaci využit framework, který aplikaci rozděluje na vrstvy, využívá zavedených vzorů, a celkově tak strukturu zdrojových souborů velmi zpřehledňuje. Rovněž tedy umožňuje současnou podobu systému snadno upravit nebo ji obohatit o nové moduly či funkce. Protože systém správy stravovacího zařízení představuje rozsáhlý komplex funkcí, který nelze v čase vyhrazeném pro přípravu bakalářské práce implementovat, byly v jejím rámci realizovány pouze moduly nezbytně nutné pro základní funkcionalitu aplikace.

Aby se mohla ověřit správná funkčnost vytvořené části systému a zároveň aplikace odhalila své nedostatky, vyzýval poslední bod zadání práce k prověření použitelnosti a přívětivosti uživatelského rozhraní. Za tímto účelem obdrželo dvanáct uživatelů odlišných profesí a technických dovedností testovací archy obsahující scénář, hodnocení práce se systémem a závěrečný průzkum určený k písemnému vyjádření testera. Testy sledovaly jak čas a správnost provedení úkolů, tak i pohyb uživatele v prostředí aplikace a četnost využití prostředků nápovědy.

Výsledky testování dokazují, že ve většině případů systém splnil požadavky kladené zadavatelem i požadavky, jež stanovil vývojář. Testeři jej ohodnotili jako moderní a přehledný a téměř všichni zvládli bez jakéhokoliv předcházejícího zaučení a bez větších obtíží splnit požadované úkoly. Jejich připomínky a návrhy byly posléze zapracovány v kapitole 5.4.

Vývoj systému *Gadmin* bude i nadále pokračovat. V první řadě se zaměří na realizaci úprav a doporučení získaných od testerů, a následně pak zahájí druhou fázi implementací zbývajících modulů, které uživateli zajistí plnou kontrolu nad sestavováním jídelníčků, správou financí a nastavením systému.

Přehled použitých zkratk a pojmů

ACL	(Access Control List) nástroj pro řízení přístupu a práv pomocí definice uživatelských rolí, zdrojů a oprávnění
AJAX	(Asynchronous Javascript And XML) technologie přenosu dat na pozadí webové stránky prostřednictvím skriptovacího jazyka JavaScript a XML
ajaxifikace	(ajaxification) proces, při němž je běžná webová stránka převedena na webovou aplikaci využívající pro zaslání HTTP požadavků AJAX mechanismus
CSS	(Cascade Style Sheet) jazyk pro popis stylu zobrazení prvků hypertextového dokumentu
DOM	(Document Object Model) objektově orientovaný model XML (potažmo HTML) dokumentu umožňující manipulaci s jeho jednotlivými prvky
drag&drop	způsob ovládání prvků uživatelského rozhraní; uchopení, následné tažení a upuštění objektu
DRY	(Don't Repeat Yourself) principy pro programování bez zbytečného opakování informací a zdrojového textu
ERA model	(Entity Relationship Attribute) abstrakt popisující vztahy mezi entitními množinami databáze
escapování	způsob zakódování znaků, které mohou být interpretovány jinak než jako znaky řetězce
fokus	stav, jenž prvek uživatelského rozhraní obdrží, je-li vybrán pro vkládání vstupní informace
framework	abstrakce sloužící jako programová podpora pro vývoj dalších aplikací

GIF	(Graphics Interchange Format) grafický formát pro rastrovou grafiku s podporou animací
GUI	(Graphical User Interface) grafické uživatelské rozhraní, vizuální prostředky (např. tlačítka, tabulky, menu, ikony) umožňující snadné ovládání aplikace
HTML	(HyperText Markup Language) značkovací jazyk sémanticky popisující hypertextové dokumenty
HTTP	(HyperText Transfer Protocol) bezstavový protokol pro komunikaci mezi serverem a klientem používaný především pro přenos webových stránek
HTTPS	(HyperText Transfer Protocol Secure) HTTP protokol šifrovaný pomocí SSL, chráněný proti odposlechu a podvržení dat
JavaScript	interpretovaný jazyk používaný především pro zajištění interakce v hypertextovém dokumentu
jQuery	JavaScriptová knihovna rozšiřující možnosti samotného jazyka, která se snaží zajistit jednotnou interpretaci JavaScriptu ve všech prohlížečích
MD5	(Message Digest Algorithm) kryptografická hašovací funkce, která ze vstupního řetězce libovolné délky vytvoří výstup o pevné délce (hash, otisk)
MVC	(Model-View-Controller) softwarově-architektonický vzor rozdělující aplikaci na tři vrstvy – datovou (model), řídicí (controller) a vrstvu uživatelského rozhraní (view)
MySQL	databázový systém poskytující komunikaci pomocí dotazů jazyka SQL
open-source	software, jenž je volně dostupný a s ním i jeho zdrojový text, který může uživatel za určitých podmínek nejen prohlížet, ale i modifikovat
NEON	formát serializace dat vyvinutý v rámci frameworku Nette, jeho syntaxe je velmi podobná YAML
PDF	(Portable Document Format) datový formát pro generování snadno přenositelných a tisku přizpůsobených dokumentů
sezení	(session) informace o uživateli doplňující kontext HTTP požadavků

PHP	programovací, od verze 5 plně objektově orientovaný jazyk pro vytváření dynamických stránek
PNG	(Portable Network Graphics) grafický formát pro rastrovou grafiku s podporou alfa kanálu
XML	(eXtended Markup Language) značkovací jazyk sémanticky popisující dokumenty
XSS	(Cross Site Scripting) útok podstrčením vlastních (zpravidla škodlivých) příkazů webovým stránkám s neošetřenými výstupy
YAML	(YAML Ain't Markup Language) formát pro serializaci dat do podoby čitelné jak člověkem, tak i počítačem
SEO	(Search Engine Optimization) principy vytváření webových stránek optimalizovaných pro internetové vyhledávače
URL	(Uniform Resource Locator) textový řetězec přesně specifikující umístění dokumentu či služby
vCard	souborový formát pro výměnu osobních dat používaný především jako elektronická podoba obchodní vizitky

Literatura

- [1] ELMASRI, R. – NAVATHE, S. B. *Fundamentals of Database Systems*. 6. vydání. USA: Addison-Wesley Publishing, 2010. ISBN-10: 0136086209, ISBN-13: 978-0136086208.
- [2] *PHP: show_source – Manual* [online]. ©2001–2013 The PHP Group, poslední změna 19.4.2013 [cit. 24.4.2013]. Dostupné z: <http://php.net/manual/en/function.show-source.php>
- [3] *PHP: mysql_escape_real_string – Manual* [online]. ©2001–2013 The PHP Group, poslední změna 19.4.2013 [cit. 25.4.2013]. Dostupné z: <http://php.net/manual/en/function.mysql-real-escape-string.php>
- [4] GROSS, J. *Authentication and Authorization* [online]. 13.3.1997 [cit. 7.3.2013]. Dostupné z: <http://www.acm.uiuc.edu/workshops/security/auth.html>
- [5] *Srovnání parametrů* [online]. ©2013 Freehosting ENDORA.cz [cit. 8.4.2013]. Dostupné z: <http://endora.cz/vlastnosti>
- [6] FRIEDL, Stephen. *SQL Injection Attacks by Example* [online]. 27.12.2004, poslední změna 10.11.2007 [cit. 14.12.2012]. Dostupné z: <http://www.unixwiz.net/techtips/sql-injection.html>
- [7] Surprer. *PHP Injection* [online]. 11.6.2006 [cit. 14.12.2012]. Dostupné z: <http://www.soom.cz/index.php?name=usertexts/show&aid=284&title=PHP-Injection>
- [8] *Thin Client Definition* [online]. ©2013 TechTerms.com [cit. 24.4.2013]. Dostupné z: <http://www.techterms.com/definition/thinclient>
- [9] *Zabezpečení před zranitelnostmi* [online]. ©2008–2013 Nette Foundation [cit. 25.4.2013]. Dostupné z: <http://doc.nette.org/cs/vulnerability-protection>

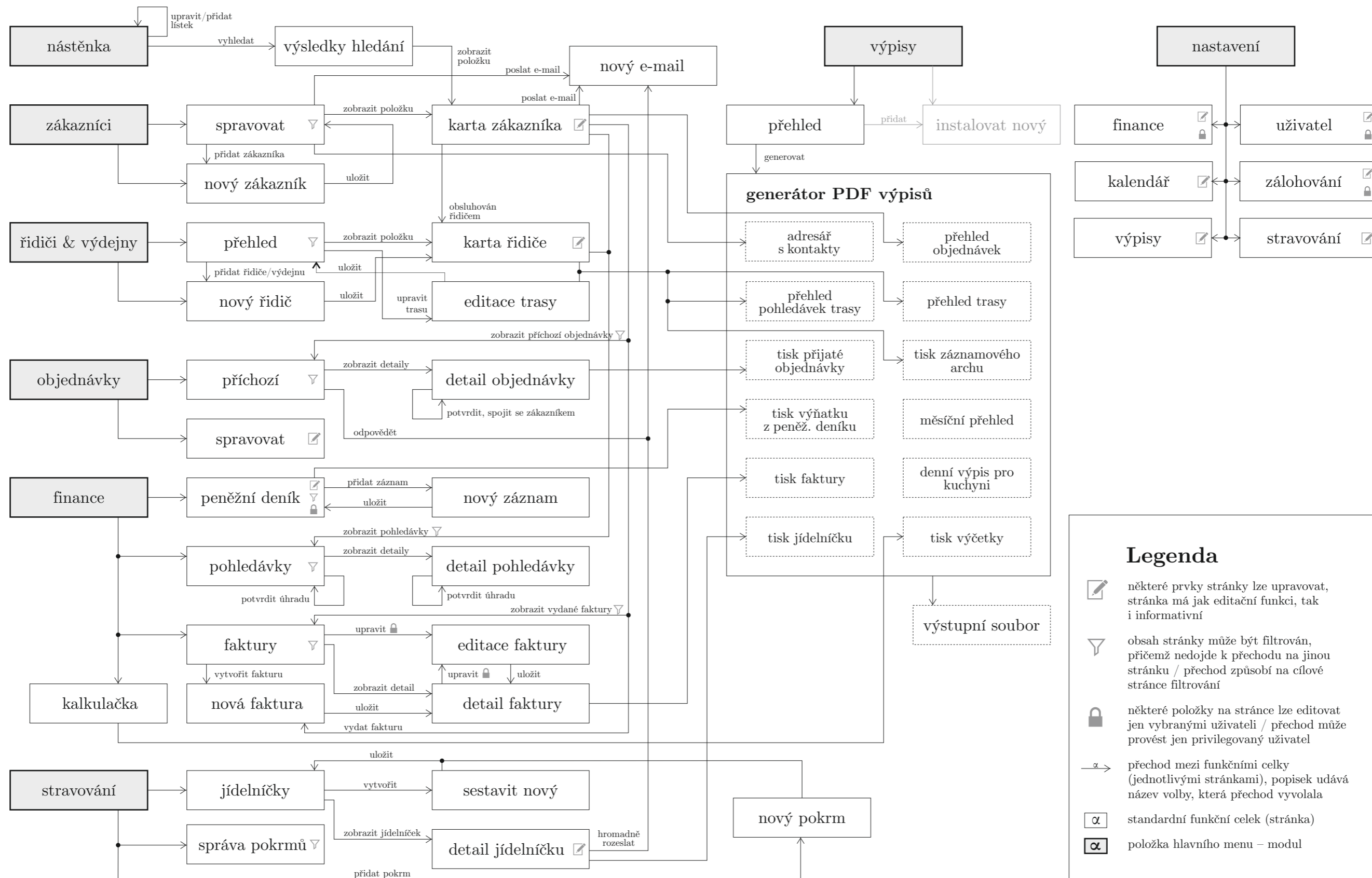
- [10] *Databáze & ORM* [online].
©2008–2013 Nette Foundation [cit. 29.4.2013].
Dostupné z: <http://doc.nette.org/cs/database>
- [11] *About jQuery UI* [online].
©2013 jQuery Foundation [cit. 29.4.2013].
Dostupné z: <http://jqueryui.com/about/>
- [12] *HTML Global title Attribute* [online].
©1999–2013 Refsnes Data [cit. 26.4.2013].
Dostupné z: http://www.w3schools.com/tags/att_global_title.asp
- [13] McARTHUR, K. *Pro PHP: Patterns, Frameworks, Testing and More*.
USA: Apress, 2008.
ISBN-10: 1590598199, ISBN-13: 978-1590598191.
- [14] GRUDL D. *Nette Framework: MVC & MVP* [online].
24.3.2009 [cit. 26.4.2013].
Dostupné z: <http://www.zdrojak.cz/clanky/nette-framework-mvc-mvp>
- [15] *Routování URL* [online].
©2008–2013 Nette Foundation [cit. 26.4.2013].
Dostupné z: <http://doc.nette.org/cs/routing>
- [16] *MVC aplikace & presentery* [online].
©2008–2013 Nette Foundation [cit. 27.4.2013].
Dostupné z: <http://doc.nette.org/cs/presenters>
- [17] *Šablony* [online].
©2008–2013 Nette Foundation [cit. 27.4.2013].
Dostupné z: <http://doc.nette.org/cs/templating>
- [18] ZAKAS N. C. – McPEAK J. – FAWCETT J. *Professional Ajax*.
2. vydání. USA: John Wiley & Sons, 2007.
ISBN-10: 0470109491, ISBN-13: 978-0470109496.
- [19] *jQuery.ajax()* [online].
©2013 jQuery Foundation [cit. 29.4.2013].
Dostupné z: <http://api.jquery.com/jQuery.ajax>
- [20] *AJAX & snippety* [online].
©2008–2013 Nette Foundation [cit. 29.4.2013].
Dostupné z: <http://doc.nette.org/cs/ajax>
- [21] HLÍNA I. *Statické ACL v modulárnej aplikácii* [online].
©2008–2013 Nette Foundation, poslední změna 1.2.2013 [cit. 29.4.2013].
Dostupné z: <http://pla.nette.org/cs/staticke-acl>

- [22] COYIER C. *All About Floats* [online].
8.7.2009 [cit. 1.5.2013].
Dostupné z: <http://css-tricks.com/all-about-floats>
- [23] *CSS3 Properties* [online].
©2000–2013 Quackit.com [cit. 1.5.2013].
Dostupné z: <http://www.quackit.com/css/css3/properties>
- [24] KYRNIN J. *What Are CSS Vendor Prefixes* [online].
©2013 About.com [cit. 1.5.2013].
Dostupné z: <http://webdesign.about.com/od/css/a/css-vendor-prefixes.htm>
- [25] COYIER C. *Data URIs* [online].
25.3.2010 [cit. 1.5.2013].
Dostupné z: <http://css-tricks.com/data-uris>
- [26] *DataTables – API* [online].
©2008–2013 SpryMedia [cit. 1.5.2013].
Dostupné z: <http://www.datatables.net/api>
- [27] *Browsers Statistics* [online].
©1999–2013 Refsnefs Data [cit. 6.5.2013].
Dostupné z: http://www.w3schools.com/browsers/browsers_stats.asp

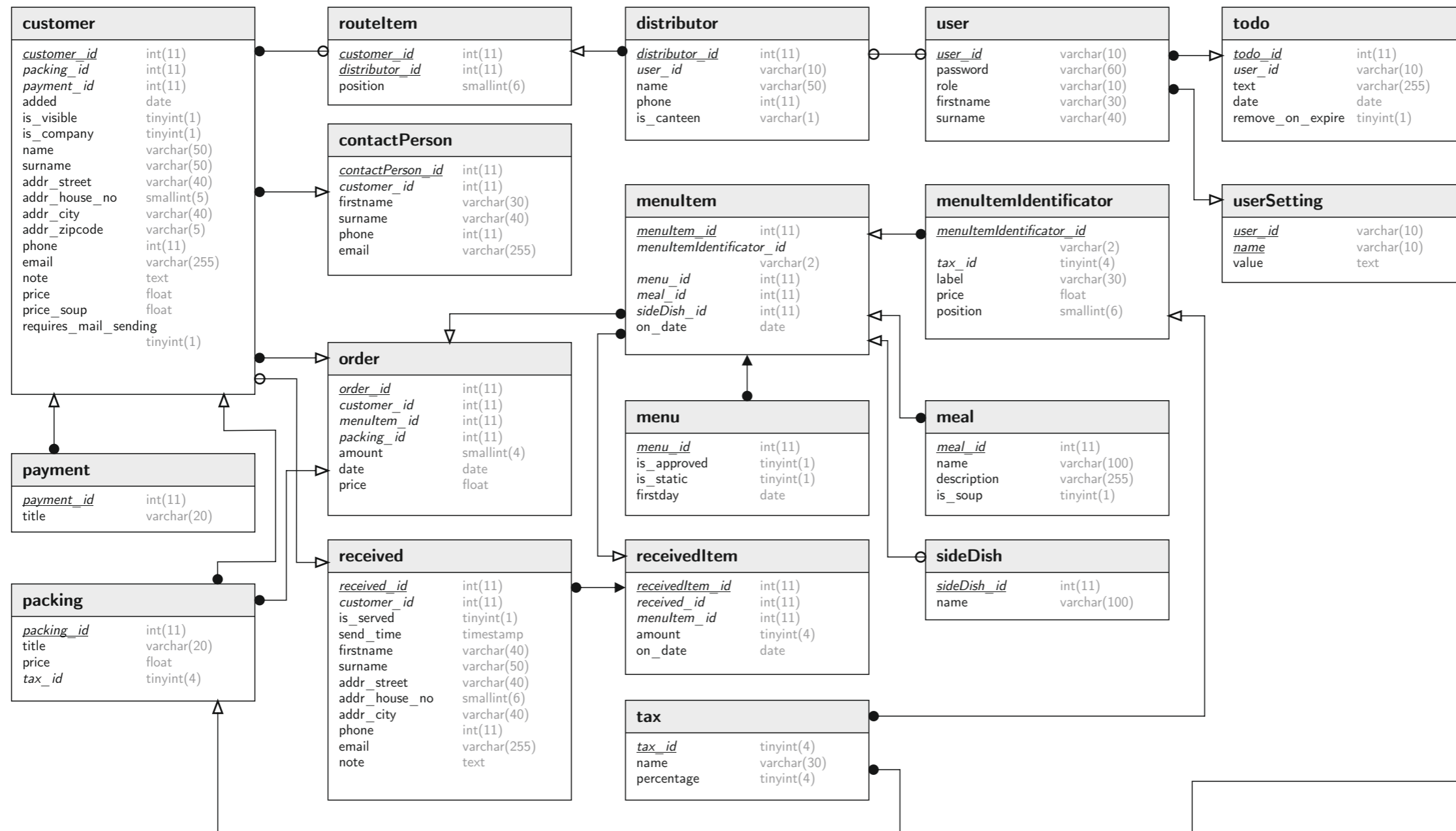
Přílohy

- A Diagram funkčních celků systému **Gadmin**
- B Návrh databáze systému **Gadmin**
- C Testovací arch
- D Testy kompatibility
- E Časový průběh testů
- F Hodnocení GUI
- G Graf učení
- H Posudek zadavatele
- I Obsah přiloženého CD

A Diagram funkčních celků systému Gadmin



B Návrh databáze systému Gadmin



Legenda

Bachmanova notace

- 0..1 —○
- 1 —●
- 0..N —▷
- 1..N —▶

Označení klíčů

- primární klíč
- cizí klíč
- standardní sloupec

Tento ERA model popisuje pouze tu část databáze, jež je využívána moduly aplikace implementovanými v rámci bakalářské práce, nikoliv kompletní databázi.

C Testovací arch

Aplikace Gadmin, kterou v následujících chvílích budete testovat, slouží ke snadné správě stravovacího zařízení, tj. umožňuje obsluhu přijímat a zpracovávat objednávky, definovat či editovat trasy distributorů a v neposlední řadě uživateli poskytuje kompletní přehled nad jeho zákazníky.

Předem děkuji za Vaši spolupráci.
Michal Kacerovský

Informace o testování

Jméno a příjmení testera

Datum testování

Pokyny pro testera

1. Testovaná verze aplikace je plně optimalizována **pouze** pro prohlížeč **Google Chrome**. Ačkoliv ji lze spustit i v jiných prohlížečích, není v nich zaručena 100% kompatibilita, a proto prosím použijte výhradně zmíněný **Google Chrome**.
2. Před samotným testováním budete vyzváni k seznámení se s prostředím aplikace. Nezhleďte prosím tento požadavek, skutečně si projděte jednotlivá menu a pokuste se porozumět jednotlivým funkcím.
3. Při testování je důležité, abyste se věnovali pouze instrukcím v testovacím archu a nebyli nuceni testy přerušit. Při testování se zaznamenává veškerá Vaše interakce s aplikací a s ní i časové údaje. Pauzy mezi jednotlivými kroky testování by tedy mohly výsledky znehodnotit. Ujistěte se proto prosím, že testy nebudete muset přerušit, a pokud se tak stane, vyžádejte si prosím nový testovací arch a proveďte je znovu.
4. Pokud máte zkušenosti s ovládáním předchozí verze aplikace (označována jako **Správce**), pokuste se při testování sledovat, jak se jednotlivé části systému změnilo, zda je jejich ovládání nyní pohodlnější či naopak.
5. Při hodnocení používejte hodnoty 1–5 jako známky ve škole. Při určování náročnosti konkrétního kroku berte v potaz, že hodnota 1 znamená nejnižší obtížnost.

Profil testera

Pro vyhodnocení výsledků je důležité vědět, jaké jsou Vaše dovednosti pro práci s počítačem a jak často s programy charakterně podobnými Gadminu pracujete. Zaškrtněte prosím pravdivá tvrzení.

Umím:

- vytvořit jednoduchý dokument v textovém editoru,
 - vytvořit tabulku s použitím základních funkcí (např. SUMA, PRŮMĚR) v tabulkovém procesoru,
 - zmenšit fotografii a upravit její barvy v grafickém editoru,
 - komunikovat prostřednictvím instant-messengerů (např. Skype),
 - odesílat a přijímat e-maily,
 - používat online nástroje, jako jsou např. Google Docs, Mapy.cz, Aukro.cz.
 - používat klávesové zkratky pro práci s prvky formuláře (Tab, Shift+Tab)
-
- V minulosti jsem pracoval(a) s předchozí verzí systému,
 - každý den využívám internetových služeb,
 - dokáži rozlišit a popsat pojmy tlačítko, odkaz, menu a zaškrtačací pole.

Jsem:

- student / nezaměstnaný
- zaměstnanec stravovacího zařízení
- zaměstnanec nebo podnikatel v jiném odvětví

Spuštění aplikace

Otevřete prohlížeč Google Chrome a přejděte na adresu

<http://varimevam.cz/gadmin-playground>

Zobrazí se úvodní obrazovka aplikace. Jako přihlašovací údaje použijte jméno **tester** a heslo **test**. Klepnutím na **přihlásit** vstoupíte do systému.

Seznámení s aplikací

Následující část **není** součástí testování, pouze Vám dává prostor pro seznámení se s aplikací. Prohlédněte si jednotlivá menu, prozkoumejte jejich obsah a význam, vyzkoušejte funkcionalitu ovládacích prvků. V tuto chvíli můžete v aplikaci provádět libovolné změny.

Ke snadnější orientaci v systému **Gadmin** by Vám měla pomoci kontextová nápověda, kterou jsou opatřeny téměř všechny podstatné prvky – podržíte-li na malý okamžik kurzor nad vybraným objektem, zobrazí se černá bublina se stručným popisem významu prvku nebo s detaily, které se k dané položce vztahují.

Aplikace je také vybavena **interaktivním průvodcem**, jenž má seznámení se systémem zjednodušit a urychlit. Pokud aktuálně zobrazené menu obsahuje tohoto průvodce, zobrazí se na liště umístěné pod hlavním menu tlačítko **tipy & triky**. Klepnutím na něj otevřete v pravém dolním rohu obrazovky průvodce s tipy. Prvek, k němuž se text vztahuje, vždy aplikace označí červeným rámečkem. Tipy & triky můžete kdykoliv zobrazit i během testování.

Až budete mít pocit, že se v aplikaci orientujete, přejděte na adresu

<http://varimevam.cz/gadmin>

Aplikace se Vás zeptá, zda chcete ze stránky skutečně odejít. Vyberte **Opustit tuto stránku**. Nyní se nacházíte ve verzi aplikace určené k testování.

Část I / Nový zákazník

Stravovací zařízení získalo nového zákazníka. Ten si přeje odebírat pokrmy v **kastrůlcích** a platit je **převodem**. S tímto zákazníkem jste se dohodli na snížení ceny polévky o 20 %, tj. její cena bude činit 16 Kč. **Vložte zákazníka** na základě uvedených údajů do databáze.

VŠEZEP, s.r.o. – Hájek 35, 345 06 Všeruby
Tel.: +420 379 562 311

Zaměstnanec této společnosti, Petr Vondráček, který s Vámi odběr obědů vyjednával, telefonoval z čísla +420 774 153 339, uvedl také svoji e-mailovou adresu **vondracek.petr@centrum.cz**. **Připojte tento kontakt** k zákazníkovi VŠEZEP, s.r.o., vytvořenému v předchozím kroku.

přehlednost formuláře nový zákazník	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
vkládání kontaktní osoby k zákazníkovi	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
celková náročnost části I	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5

Část II / Distribuce

Nově přidaný zákazník VŠEZEP, s.r.o, se sice nyní nachází v databázi zákazníků, ale zatím jej nikdo neobsluhuje – není zahrnut v trase některého z distributorů, proto **přidejte nového řidiče** s Vaším jménem a smyšleným telefonním číslem. Tohoto řidiče propojte s uživatelským účtem, k němuž jste nyní přihlášení (Pavel Novák). Z trasy řidiče Ondra **odeberte následující zákazníky**:

- Tesařík František
- SANO, s.r.o.
- Pečený Jan

Zmíněné zákazníky **vložte do trasy** řidiče s Vaším jménem. Přidejte k nim také zákazníka VŠEZEP, s.r.o. (Na pořadí zákazníků v trase nezáleží.)

přehlednost formuláře nový distributor	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
editace trasy	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
celková náročnost části II	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5

Část III / Objednávky

Zákazník VŠEZEP, s.r.o., si na úterý 23. dubna 2013 přeje objednat pokrmy uvedené v tabulce níže. Nepřehlédněte, že si zákazník u poslední položky vyžádal změnu balení! Po jejich vložení do databáze ověřte na kartě zákazníka v rámečku **Statistiky**, že jeho celková útrata nyní činí 340,00 Kč (za 5 polévek a 5 pokrmů).

5×	polévka	Kulajda
3×	pokrm 1	Vepřová krkovice po selsku, houskový knedlík
2×	stálý pokrm 2	Smažený vepřový řízek, brambory (zabalit do vaxu)

práce s editorem objednávek	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
způsob změny balení a ceny u konkrétního pokrmu	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
celková náročnost části III	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5

Část IV / Příchozí objednávky

Gadmin po celou dobu signalizuje, že v menu **Objednávky** čekají na odbavení 3 příchozí požadavky. Zobrazte příchozí objednávku od **Amálie Královcové**. V současné chvíli není možné objednávku schválit, protože systém osobu s tímto jménem nenalezl v databázi. Proto je nutné, abyste tuto novou zákaznici **přidali do systému** (lze jedním kliknutím).

Jakmile paní Královcovou přidáte do databáze a vrátíte se k její objednávce, Gadmin Vás upozorní, že pravděpodobně rozpoznal odesilatele a nabídne Vám jeho spojení s touto objednávkou. Ve chvíli, kdy je příchozí objednávka propojena se zákazníkem, můžete její položky vložit do databáze. V případě paní Královcové tedy potvrdíte objednávku boloňských špaget. Zobrazí se výstraha – **Královcová Amálie** totiž není zařazena do žádné z tras, proto ji přiřaďte k některému z distributorů podobně jako v předchozí části.

Následně přijměte také objednávky **Tomanové Zdeňky** a **Pečeného Honzy**. Poté pomocí editoru objednávek **změňte balení** objednávky **Zdeňky Tomanové** (dne 29. dubna, řidič **Ondra**) na **vax** a **cenu** upravte na **35,00 Kč**. Stejná částka by se pak měla objevit v celkové útratě na kartě tohoto zákazníka.

automatické přidání zákazníka na základě údajů z příchozí objednávky	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
<hr/>										
příjem (schválení) příchozí objednávky	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
<hr/>										
celková náročnost části IV	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5

Dokončení testování

Abyste úspěšně dokončili testování, klepněte v uživatelském menu (jméno **Pavel Novák** v horní části aplikace) na položku **Ukončit testování**. Aplikace vygeneruje soubor se souhrnem Vaší interakce během testování a nabídne jej ke stažení. Uložte jej a odešlete na e-mail **kacerov2@students.zcu.cz**.

Poznámka Může se stát, že systém nenabídne soubor ke stažení, ale zobrazí jej přímo v prohlížeči. V takovém případě prosím klepněte pravým tlačítkem a vyberte možnost **Uložit jako....** Následně jej odešlete na uvedený e-mail.

Závěrečný průzkum

Následující otázky sledují, jak se Vám se systémem Gadmin pracovalo, a poskytují Vám možnost vyjádřit se k jeho součástem písemnou formou.

Jak na Vás působí vzhled aplikace?

.....

Je hlavní menu a podmenu dobře koncipováno? Provedli byste v něm změny?

.....

.....

Je průvodce tipy&triky užitečný? Pomohl Vám lépe se orientovat?

.....

Rozuměli jste všem funkcím a ikonám? Pokud ne, kterým?

.....

.....

Které funkce Vám připadají užitečné a myslíte si, že najdou své využití?
Které naopak ne? Postrádáte nějakou v rámci implementovaného rozsahu?

.....

.....

.....

Zde můžete vyjádřit Váš názor na aspekty, jež nebyly zmíněny.

Co je podle Vás největším přínosem v nové verzi systému oproti staré?
(odpoví pouze ti, co se starou verzí pracovali)

.....

.....

.....

.....

.....

.....

.....

D Testy kompatibility

		Google Chrome 26.0	Opera 12.15	Mozilla Firefox 20.0	Internet Explorer 9.0	Internet Explorer 7.0
uživatelské rozhraní	dodržení layoutu	✓	✓	✓	✓	×
	kulaté rohy rámečku	✓	✓	✓	✓	×
	barevný přechod tlačítek	✓	✓	✓	×	×
	vnitřní/vnější stín	✓	✓	✓	✓	×
	stín textu	✓	✓	✓	×	×
	animace CSS vlastností	✓	✓	✓	×	×
	průhlednost	✓	✓	✓	✓	×
	grafika s alfa kanálem	✓	✓	✓	✓	✓
	vytažení objektu (outline)	✓	✓	✓	×	×
dynamika	ajaxifikace odkazů	✓	✓	✓	✓	×
	ajaxifikace formulářů	✓	✓	✓	✓	×
	ostatní AJAX	✓	✓	✓	✓	×
	plugin dataTables	✓	✓	✓	✓	✓
	plugin powerTip	✓	✓	✓	✓	! ³
	plugin dayPicker	✓	✓	✓	✓	×
	editor objednávek	✓	! ¹	✓	✓	×
	centrování aktiv. pole editoru	✓	✓	✓	✓	×
	editace položek trasy	✓	✓	✓	✓	! ⁴
	filtrace položek editoru trasy	✓	✓	✓	×	×
	blokování navigace v prohlížeči	✓	×	✓	✓	✓
	průvodce tipy&triky	✓	✓	✓	! ²	×
	obnovení přihlášení	✓	✓	✓	✓	×
	funkce nahoru	✓	×	×	×	×
placeholder vstupu	✓	✓	✓	×	×	
podpora systému Gadmin	100 %	90 %	96 %	69 %	17 %	

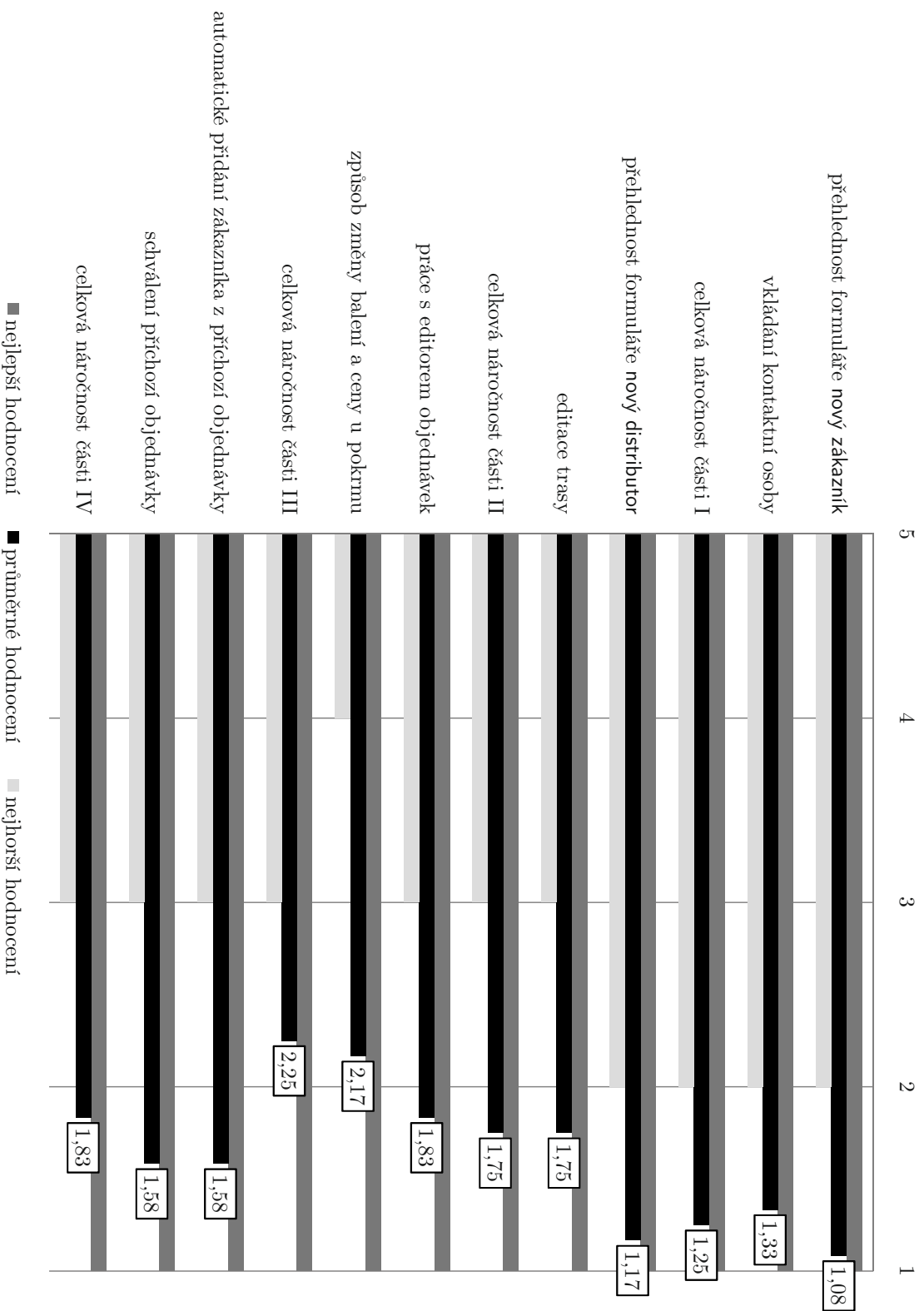
¹ ukládá i vstupy, jejichž hodnota nebyla změněna; pouze získaly a ztratily fokus

² při změně typu nedochází ke správnému zarovnání boxu do pravého dolního rohu

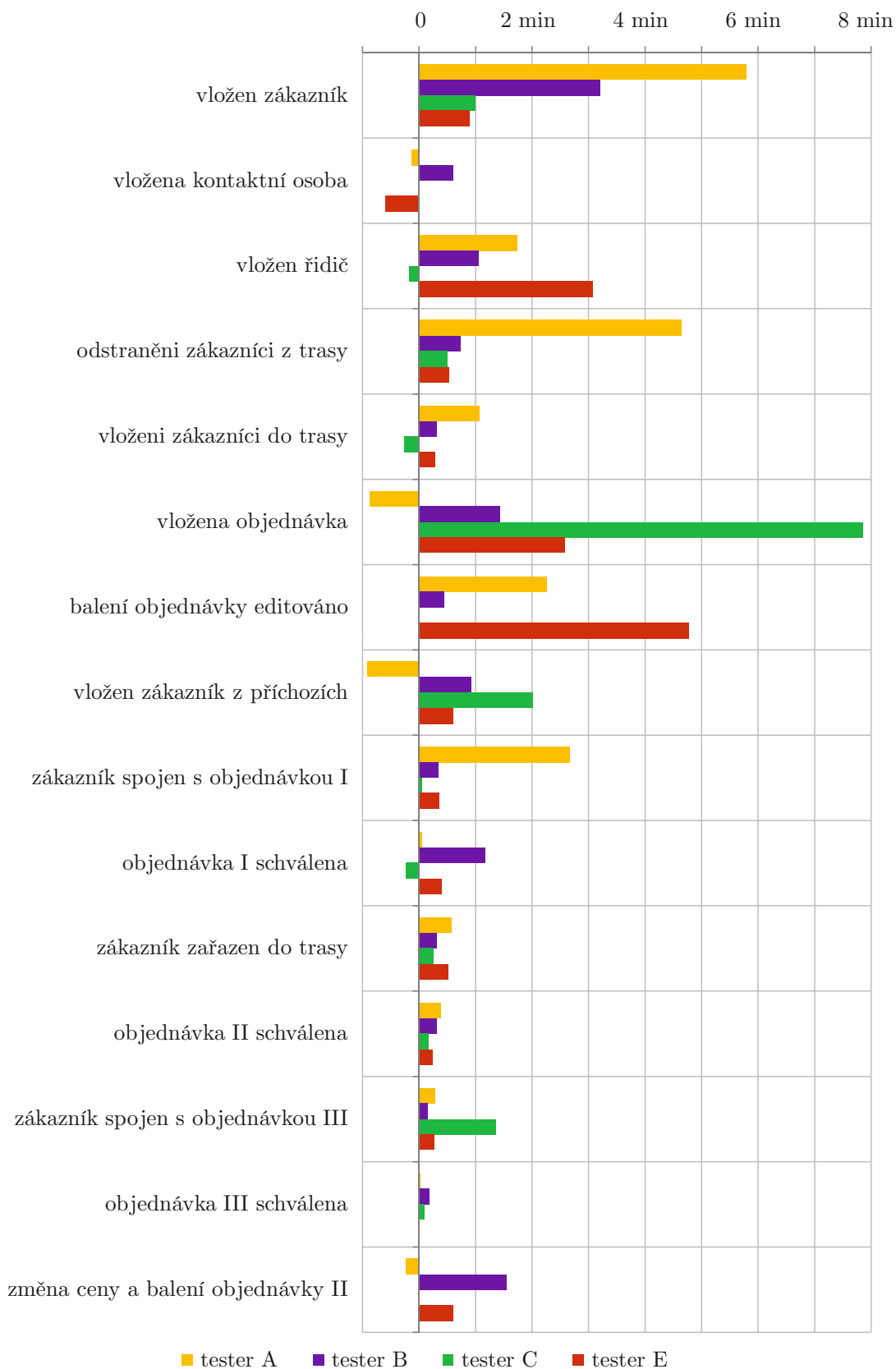
³ nezobrazuje šipku směřující na prvek, k němuž je kontextová nápověda vázána

⁴ nesprávně zobrazen, drop efekt má neobvyklé chování (nebrání správné funkcionalitě)

F Hodnocení GUI



G Graf učení



H Posudek zadavatele

Už původní program mi pomohl vyřešit spoustu starostí a problémů, avšak postupem času se objevily další. Ty největší však spočívaly v placení obědů, protože má naše společnost mnoho zákazníků, u nichž se cena může lišit (např. v závislosti na způsobu balení odebíraných pokrmů). Nová verze této aplikace nám však umožní spočítat celkovou tržbu za určité období mnohem snadněji. Umožní přičítat k ceně oběda i náklady na jeho balení, což jsem byla doposud nucena dělat ručně, na papíře.

Obzvláště nadšená jsem z automatického vkládání příchozích objednávek. Tato funkce mi ušetří hodně času, protože nebudu muset každý den sčítat objednávky z e-mailů přijatých od zákazníků.

I celkový pohled na aplikaci je nyní mnohem lepší. Zlepšila se přehlednost jednotlivých panelů a také jejich funkcionalita prošla řádnými změnami. Velice praktický mi připadá průvodce tipy&triky, který si mohu zobrazit, když zrovna potřebuji. Rovněž oceňuji informaci o tom, kdo aktuálně slaví svátek, a funkci nástěnky, která mi bude připomínat důležité úkoly, na něž bych mohla jinak snadno zapomenout. Také ikonky, které přímo v objednávkě signalizují, jakým způsobem bude pokrm balen, jsou bez pochyby usnadněním a zrychlují orientaci.

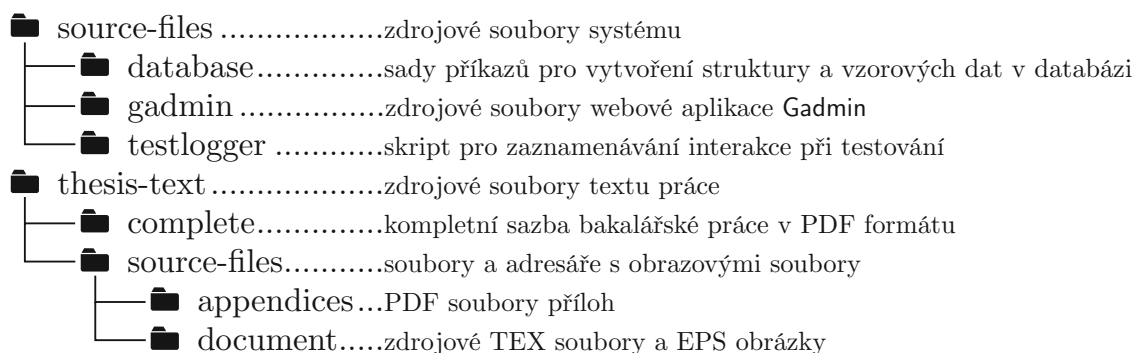
Určitě existuje mnoho věcí, které bych pochválila, ale nedokážu si všechny vybavit najednou, pravděpodobně je však ocením při nasazení systému do provozu. Tak rozmanitý objednávkový systém, který naše stravovací zařízení nyní má, nebyli schopni vytvořit ani specializovaní odborníci z Prahy, kteří mě nutili připravovat si do jejich aplikace tabulky v Excelu, a to včetně vzorců. Za tuto radu si účtovali nemalou částku. Dlouho jsme hledali program, který by zvládal všechny požadované funkce, a ani jsme nedoufali, že jej někdo dokáže vytvořit. Nyní jsme ho našli, to je on.

Hana Marková

administrativní pracovnice
stravovacího zařízení

I Obsah přiloženého CD

Adresářová struktura disku



Zprovoznění systému

Systém Gadmin je webová aplikace, pro jejíž instalaci musí prostředí webového serveru splňovat minimální požadavky frameworku **Nette**¹ a poskytovat databázový systém MySQL, včetně typu úložiště InnoDB. Gadmin je plně kompatibilní s aktuální verzí prohlížeče **Google Chrome**. Chcete-li jej spouštět v jiném prohlížeči, ověřte nejdříve, do jaké míry systém podporuje (viz příloha D).

1. Přeneste obsah složky `/source-files/gadmin` do adresáře webového serveru, z něhož má být Gadmin dostupný.
2. Do databáze importujte SQL soubory ze složky `/source-files/database` v pořadí 1 - `struktura.sql`, 2 - `data.sql` a nakonec 3 - `integrita.sql`. Import všech tří souborů v uvedeném pořadí je **nezbytně nutný** pro správný chod aplikace.
3. Z adresáře instalace přejděte do `/app/config` a otevřete soubor `config.neon`. Je-li potřeba, upravte přístupové údaje k databázi (podrobněji se konfigurací v textu práce věnuje kapitola 4.2.6). Ujistěte se, že v souboru nedošlo k nahrazení znaků tabulátoru mezerami, a uložte jej.
4. Gadmin je nyní plně funkční. K přístupu do systému použijte uživatelské jméno `tester` a heslo `test`.

¹Požadavky frameworku jsou dostupné z: <http://doc.nette.org/cs/requirements>