

Univerzita Pardubice
Fakulta elektrotechniky a informatiky

Nemocniční objednávkový systém

Michal Seifert

Bakalářská práce
2008

Souhrn

Práce je zaměřena na problematiku nemocničního objednávkového systému. V teoretické části jsou rozebrány webové technologie s cílem vybrat neoptimálnější řešení pro vývoj aplikace. Dále je provedena analýza současných objednávkových systémů se zaměřením na systémy používané ve zdravotnictví. V praktické části je vytvořeno webové sídlo Chrudimské nemocnice, a.s. s objednávkovým modulem a možností správy celého systému přes administrátorské rozhraní.

Klíčová slova

Nemocniční objednávkový systém, rezervační systém, webová prezentace, PHP, MySQL.

Title

Hospital Ordering System.

Abstract

The work is focused on issues of a hospital ordering system. The theoretical part analyses web technologies with an aim to choose the most optimal solution for application development. Further, an analysis of current ordering systems is done with a focus on systems used in healthcare. In the practical part, the web site of the hospital Chrudimska Nemocnice, A.S. is created, including an order module and possibilities of administrating the whole system through an administrator interface.

Keywords

Hospital Ordering System, reservation system, web presentation, PHP, MySQL.

Obsah

1. Úvod.....	10
2. Přehled webových technologií a norem.....	11
2.1 Programovací a značkovací jazyky.....	11
2.2 Databázové systémy.....	15
2.3 Webové servery.....	16
2.4 Výběr nástrojů pro vývoj.....	17
3. Projekt.....	18
3.1 Charakteristika klienta.....	18
3.2 Stanovení cílů projektu.....	18
3.3 Způsoby zavedení objednávkového systému.....	20
3.4 Analýza objednávkového systému.....	22
3.5 Architektura objednávkového systému.....	26
3.6 Návrh databáze.....	27
4. Implementace objednávkového systému.....	32
4.1 Adresářová struktura zdrojových souborů.....	32
4.2 Knihovny funkcí.....	32
4.3 Bezpečnost aplikace.....	36
4.4 Základní funkcionalita aplikace.....	37
5. Závěr.....	40

Seznam obrázků

Obr. 1: Komunikace mezi klientem a serverem.....	16
Obr. 2: Hierarchické zobrazení aktérů systému.....	22
Obr. 3: Digram užití systému.....	23
Obr. 4: Architektura systému.....	27
Obr. 5. ERD diagram první část.....	28
Obr. 6: ERD diagram druhá část.....	29
Obr. 7: Administrátorské rozhraní.....	38
Obr. 8: Vnější vzhled webové prezentace.....	39

Seznam tabulek

Tab. 1: Detailní popis tabulky pracoviste.....	30
Tab. 2: Detailní popis tabulky pracovní údaje.....	31
Tab. 3: Detailní popis tabulky menu.....	31
Tab. 4: Detailní popis tabulky volby_nazvy.....	31
Tab. 5: Detailní popis tabulky volby_hodnoty.....	31

Seznam zkratek

XML	eXtensible Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol over Secure Socket Layer
SOAP	Simple Object Access Protocol
WSDL	Web Services Description Language
WWW	World Wide Web
HTML	HyperText Markup Language
JavaScript	multiplatformní, objektově orientovaný skriptovací jazyk
Java	objektově orientovaný programovací jazyk
PHP	skriptovací programovací jazyk
W3C	World Wide Web Consortium
SSL	Secure Socket Layer
HTML	HyperText Mark Language
SGML	Stardard Generalized Markup Language
CSS	kaskádové styly
DTD	Document Type Definition
DOM	Document Object Model
XSLT	eXtensible Stylesheet Language Transformations
ASP.NET	Active Server Pages
DHTML	Dynamic HTML
GNU	<i>GNU's Not Unix</i>

1. Úvod

Každým rokem stoupá počet uživatelů Internetu. Lidé si zvykli na komfort, který jim přináší a jsou nakloněni každé další službě, jež jim může usnadnit život. Celé kouzlo tkví v neustále se měnícím toku informací, dostupném prakticky odkudkoliv. K tomu, aby v dnešní době moderní člověk dobře nakoupil, objednal se k lékaři a zjistil, co všechno bude potřebovat na dovolenou do zahraničí, už nepotřebuje tolik času a energie jako dříve. Stačí se připojit na Internet a vše je během chvilky zařízeno. Nemusíme nikam chodit, čekat ve frontě, vynakládat spoustu energie a finančních prostředků úplně zbytečně. Internet nám tedy tvoří ideálního prostředníka mezi zákazníkem a poskytovatelem služeb. Je proto vhodné, pokud to daná situace umožňuje, využít těchto možností a ulehčit sobě i svým zákazníkům mnoho starostí.

Cílem této práce je navrhnout a implementovat nemocniční objednávkový systém, který by měl být pro pacienty další alternativou, jak si zajistit rychle a přehledně návštěvu u svého lékaře. Vyhnout se zbytečnému čekání a personálu ušetřit čas strávený telefonickým objednáváním nebo přímým zadáváním objednávek do systému.

2. Přehled webových technologií a norem

Při tvorbě webu máme na výběr z mnoha technologií a norem, které můžeme použít. Mnohdy nám zvolení nevhodného nástroje může přidělat spoustu problémů. V tomto přehledu jsou stručně popsány technologie, mezi kterými jsem se snažil vybrat ty nejvhodnější pro tvorbu nemocničního objednávkového systému. Vývoj webových technologií jde neustále kupředu a přetváří tak požadavky kladené na webové aplikace. Dřívější řešení speciálních programů, které byly často závislé na operačním systému klienta a většinou se úzce specializovaly na řešení jednoho problému, jsou už pro dnešní dobu nevyhovující. Trend se ubírá směrem k aplikacím, které budou fungovat na nejrůznějších zařízeních a systémech, data v nich prezentovaná budou neustále aktuální a přístup k těmto informacím bude možný odkudkoliv a kdykoliv. Jedno z možných řešení tohoto problému jsou aplikace založené na principu klient - server.

Klient, zastoupený webovým prohlížečem, vznesе požadavek na server a ten mu vrátí odpověď, kterou je nejčastěji (X)HTML stránka doplněná třeba o Javascript nebo formátování pomocí kaskádových stylů. Webových serverů je k dispozici celá řada a umějí komunikovat s klientem pomocí různých protokolů, nejčastěji využíván HTTP protokol. K uchování dat se pak využívají databázové systémy. V této kapitole jsem čerpal z těchto zdrojů [1], [2], [3], [4], [5], [7], [8], [9] .

2.1 Programovací a značkovací jazyky

Bez programovacích a značkovacích jazyků se neobejde žádná webová aplikace. K výběru jich je celá řada, každý z nich nabízí jiné možnosti a nelze proto jednoznačně říci, jaký je nejlepší.

Prostředky statického webu

Slouží k tvorbě statických stránek, které byly populární především v minulosti, kdy se od webu požadovala pouze jednoduchá prezentace malého rozsahu bez možnosti interakce s uživatelem.

SGML

Je metajazyk k definování různých značkovacích jazyků. Jeho smyslem bylo oddělit data od jejich zpracování a umožnit komunikaci mezi různými informačními

systemy na různých počítačových platformách. Vznikl z iniciativy velkých amerických firem. DTD definuje pouze syntaxi (jména elementů a jejich používání), nikoli sémantiku (jejich význam – tu už musí znát autor). Prohlížeč zase musí vědět, jak jednotlivé elementy zobrazit. Nedostatkem SGML bylo, že nebyl použitelný v podmínkách webu.

HTML

Místo SGML jako celku se v podmínkách internetu rozšířila jen jeho aplikace HTML. Základem HTML je doporučení W3C označované jako HTML 3.2. Poslední verzi HTML byla verze 4, kterou se vývoj definitivně završil. HTML se již dále nerozvíjí, pokračuje se vývojem XHTML neboli rozšířeného HTML, které je ve svých požadavcích mnohem přísnější než původní HTML.

XHTML

Nástupce HTML. Je základní značkovací jazyk pro tvorbu hypertextových dokumentů. Využívá tagy ke strukturování textu do nadpisů, sloupců, apod. Umožňuje nastavení třech typů dokumentu Strict, Transitional a Frameset. Striktní XHTML neobsahuje žádné atributy sloužící k formátování textu. Tím oddělí informace od vzhledu a umožní snadnější manipulaci a úpravy ať už vzhledu nebo kódu.

XML

Obecný značkovací jazyk, který je používán především pro výměnu dat mezi různými aplikacemi. Výhoda oproti jiným řešením spočívá v tom, že kromě samotného textu nese i informaci o jeho významu. Autor webové stránky pak může použít vlastní tagy, které dokáží mnohem přesněji vyjádřit význam prezentovaných dat. Univerzálnost celého řešení nám dává možnost využít XML prakticky kdekoli, například chceme-li uložit určité informace a poté je prezentovat v několika různých formátech.

XSLT

Zajišťuje převod XML dokumentu na jiný formát, nejčastěji XHTML nebo do známých datových struktur definovaných mezinárodním koncernem W3C. K provedení transformace jsou potřeba dva soubory. První z nich obsahuje zdrojová data ve formátu XML a druhý příkazy pro transformaci v jazyce XSL.

DOM

Pomocí tohoto objektově orientovaného programátorského rozhraní můžeme přistupovat k XML nebo HTML dokumentům a upravovat libovolně jejich obsah, strukturu nebo styl. Nezávislost na programovacím jazyku umožňuje nasazení tohoto API do libovolných aplikací.

CSS

Neboli tabulky kaskádových stylů, byly vytvořeny za účelem oddělit v HTML, XHTML nebo XML dokumentu vzhled od informace. Avšak k této ideální formě jsme se zatím ještě v reálném provozu nedopracovali. Bránil tomu především komplikovaný vývoj HTML ovlivňovaný zejména nedodržením standardů na straně tvůrců prohlížečů. Do verze XHTML umožňoval jazyk popisovat nejen obsah a strukturu dokumentu, ale i způsob zobrazení, a tím odsouval CSS do postranní.

V dnešní době se ale kaskádové styly těší čím dál větší oblibě. A není se čemu divit, přináší spoustu výhod, dalších možností a vylepšení oproti použití pouze HTML formátování. Stačí například pouze jednou napsat styl určitého elementu a pro každý další element, který má být stejně zobrazen, se jen odkážeme na původní definici. Dynamická práce se styly umožní jednoduše změnit vzhled celého webu, aniž bychom museli zasahovat přímo do zdrojového kódu a přepisovat formátování jednotlivých prvků.

Validátory zdrojového kódu

Při vývoji často programátor kombinuje různé množství nástrojů, programovacích jazyků a technologií. Koncový uživatel také využívá různorodé operační systémy a prohlížeče webových stránek, které komunikují s nejrůznějšími servery. Takže je potřeba vytvořit určitá pravidla, která by mohli všichni dodržovat a tím zajistit maximální kompatibilitu vyvíjených produktů. Právě z tohoto důvodu vzniklo mezinárodní konsorcium W3C, jehož členové společně s odbornou veřejností vytváří webové standardy pro World Wide Web. W3C normy jsou pouze doporučením a i nevalidní kód může prohlížeč zobrazit tak, jak původně autor zamýšlel. Avšak nemusí tomu tak být vždy. A pokud se návštěvníkovi zobrazí stránka špatně a v každém prohlížeči úplně jinak, určitě to na oblibě a návštěvnosti webu nepřidá.

K ověření, zda je stránka opravdu v souladu s danou normou, je možné použít validátor, který danou stránku zkontroluje. Pokud se na stránce vyskytnou chyby, jsou vypsané řádky, kde došlo k rozporu s normou a krátký popis, v čem může být problém. Naopak jestliže stránka vyhovuje, obdržíme gratulaci a možnost umístit na ni odkaz informující návštěvníky o validitě testovaných stránek, kterou si mohou pomocí odkazu ihned překontrolovat.

Prostředky dynamického webu

Umožní vytvořit prostředí, které bude schopno reagovat na požadavky uživatele a měnit dle jeho voleb zobrazovaný obsah.

JavaScript

Je multiplatformní objektově orientovaný programovací jazyk, který je interpretován na straně klienta. Nejčastěji najde využití při ovládní různých interaktivních prvků, tvoření grafických efektů a ulehčení administrace. Jako pomůcka pro doladění výsledné tváře a funkčnosti webu je Javascript výborný nástroj, avšak nesmíme zapomínat, že ne vždy ho musí mít návštěvník v prohlížeči povolený, a proto se na jeho funkčnost nedá vždy spoléhat.

ASP.NET

Tvoří součást .NET Frameworku firmy Microsoft pro tvorbu webových aplikací a služeb. Je především oblíbený v komerční sféře. Nabízí programátorům pracovat v libovolném jazyce podporujícím CLR. Zjednodušuje a urychluje vývoj webových aplikací. Stránky stačí jednoduše poskládat z ovládacích prvků webu a doprogramovat jim jejich funkce. O zbytek se postará ASP.NET a výsledný produkt prezentuje jako validní XHTML kód s Javascriptem.

Java

Objektově orientovaný programovací jazyk. Vyznačující se svou nezávislostí na platformě a možností programovat internetové aplikace, jak na straně klienta, tak i na straně serveru. Díky své přenositelnosti najde využití prakticky kdekoliv. Je označován za nejpoužívanější programovací jazyk na světě.

DHTML

Označuje skupinu nástrojů umožňující dynamické změny HTML dokumentů. Skládá se z Javascriptu, CSS a někdy i DOM v prohlížečích Internet Explorer.

Využití těchto nástrojů na statické stránce umožní třeba zobrazit či skrýt část nějakého textu nebo vysunout rozbalovací menu atd.

PHP

Je skriptovací jazyk vyvíjený jako open source. Převážně se využívá na tvorbu dynamických webů. Lze jej ovšem nasadit i při výrobě desktopových aplikací. Vyznačuje se především svou jednoduchostí, obrovským množstvím zabudovaných funkcí a výbornou podporou u poskytovatelů webhostingových služeb. Díky těmto vlastnostem a nezávislosti na platformě, jak na straně serveru, tak i klienta, se stalo PHP jedním z nejpoužívanějších skriptovacích jazyků pro tvorbu dynamických webů.

2.2 Databázové systémy

Software, zajišťující komunikaci mezi aplikacemi, serverovými požadavky a uloženými daty. Oproti obyčejnému souborovému systému se od něho očekává efektivní práce s velkým množstvím dat, jejich řízení (vkládání, modifikace, odstraňování) a definice jejich struktur.

Oracle Database

Výkonné a spolehlivé řešení databázového systému, který se proslavil především v oblasti zpracování velkých objemů dat a náročnými požadavky na zabezpečení. Je šířen pod dvěma licencemi a tím umožňuje jak placené komerční nasazení do velkých projektů, tak i bezplatné použití s určitými omezeními pod GNU licenci pro menší projekty. Celkově je velice propracovaný a pro většinu webových aplikací až příliš výkonný.

MySQL

Nejpoužívanější databázový systém pro webové aplikace. Je k dispozici pod placenou komerční licenci i bezplatně pod GNU licenci. MySQL se v počátcích optimalizovalo především na rychlost a jednoduchost, to i za cenu absence různých funkcí a vlastností, kvůli kterým bylo mnohdy kritizováno. V dnešní době jsou však již tyto neduhy z větší části napraveny, takže není problém v nasazení do náročných projektů. Velkou výhodou je velmi dobrá podpora jazykem PHP. V praxi je tato kombinace velice oblíbená a dosahuje skvělých výkonů.

PostgreSQL

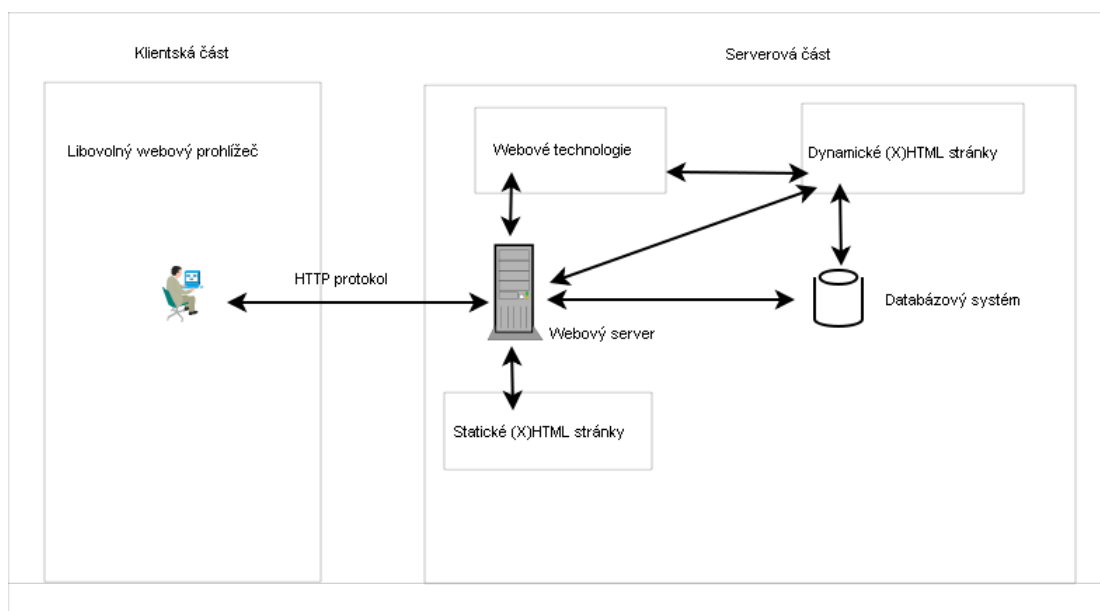
Plnohodnotný databázový systém s otevřeným zdrojovým kódem a rozsáhlou dokumentací. Nabízí širokou podporu operačních systémů a programovacích jazyků. Díky BSD licenci ho můžeme bezproblémově rozšiřovat o nové datové typy, funkce, operátory, agregační funkce a procedurální jazyky. Vhodný především pro rozsáhlé webové projekty, kde požadujeme velkou spolehlivost, bezpečnost a rozšiřitelnost.

MS SQL

Komerční produkt od firmy Microsoft. Server podporuje řadu standardů a nabízí velké množství funkcí pro práci s databází. Podobně jako Oracle je toto řešení robustní určené především pro nasazení do rozsáhlých projektů.

2.3 Webové servery

Webový server přijímá požadavky nejčastěji pomocí HTTP protokolu. Po zpracování vrací klientovi odpověď, který daný požadavek vznesl. Obsah odpovědi může být buď statická stránka nebo dynamicky generovaná informace většinou s pomocí databázového systému a programovacích jazyků na straně serveru. Součástí odpovědi je i tzv. stavový kód. Ten sděluje, jak byl požadavek proveden. Webový server často bývá doplňován o další podpůrné produkty. V závislosti na složitosti návrhu se pak využívají dva modely architektury. Prvním řešením je dvouvrstvá architektura typu klient – server, vyznačující se svou jednoduchostí. Při rozsáhlém projektu se pak využívají složitější vícevrstvé architektury, které dosahují se vzrůstajícím počtem připojených klientů lepších výsledků než dvouvrstvé.



Obr. 1: Komunikace mezi klientem a serverem

Apache

Webový multiplatformní server s otevřeným zdrojovým kódem, který má nadpoloviční podíl z celkových instalací serverů, dodávající prohlížečům webové stránky. Apache se skládá z jádra a modulů, kterými ho lze libovolně rozšiřovat. Těchto modulů je k dispozici celá řada. V případě specifických požadavků můžeme vytvářet i své vlastní. K dispozici je rozsáhlá dokumentace. Mezi nejpoužívanější moduly patří například:

Mod_PHP – modul umožňující použití jazyka PHP.

Mod_SSL – modul sloužící k šifrování přenosu.

Apache je upřednostňován především v menších a středních projektech, kde značnou roli hraje jeho bezplatná licence. Výhodou i nevýhodou může být dostupnost jeho zdrojových kódů, která na jedné straně vzbuzuje obavy, že se znalostí kódu může hacker snáze na tento server zaútočit a na druhé straně je tento kód podroben zkoumání široké veřejnosti, tudíž je snáze a rychleji odhalena chyba, než je tomu u uzavřených řešení.

IIS

Webový server od firmy Microsoft. Oblíbený především pro svou rychlost a snadnou konfigurovatelnost. K jeho běhu je vyžadován operační systém Microsoft Windows. Bývá nasazován především do větších projektů, kde se očekává použití jazyka ASP.NET. Oproti přímému konkurentovi Apache je znevýhodněn závislostí na systému Microsoft Windows a svojí cenou.

2.4 Výběr nástrojů pro vývoj

Z celé řady nástrojů, které jsou k dispozici pro vývoj webových aplikací jsem zvolil tuto kombinaci:

- **Webový server** – Apache, jsem zvolil především kvůli jeho modulárnosti, velké oblibě u poskytovatelů a otevřenému zdrojovému kódu.
- **Programovací jazyk** – PHP mě zaujal především díky své jednoduchosti a rychlosti se kterou se tvoří skripty. Jelikož tento projekt není až zas tak rozsáhlý, je PHP optimální volba. Jako doplňkový programovací nástroj jsem zvolil Javascript.
- **Databáze** – MySQL mě zaujala především kvůli dobrým zkušenostem v kombinaci s PHP, velké rozšířenosti mezi poskytovateli a bezplatné licenci.

- **Validita** – je v celém projektu dodržována dle XHTML 1.0 strict a CSS 2.1 specifikací.

3. Projekt

3.1 Charakteristika klienta

Téma bakalářské práce mi bylo zadáno firmou Chrudimská nemocnice, a.s. poskytující široké spektrum zdravotnických služeb. Speciální zákroky prováděné v centru laparoskopické chirurgie a centrum neinvazivní kardiologie jsou na takové úrovni, že jsou vyhledávány pacienti z celé České republiky. Nemocnice získala nejedno ocenění, například její porodní oddělení je od roku 1998 oceněno certifikátem BABY-FRIENDLY HOSPITAL. Jako první v České republice nabízí svým pacientům nemocniční rádio, které má přispět k zpříjemnění jejich pobytu.

3.2 Stanovení cílů projektu

V důsledku zkvalitnění služeb je Chrudimská nemocnice, a.s. stále více vyhledávána pro zajištění nejrůznějších zákroků. Aby zachovala minimálně stejnou úroveň nabízených výkonů, je nutné se rozvíjet a nezaostat za konkurencí. S narůstajícím počtem pacientů se zvyšují i požadavky na už tak vytížený personál, dostupnost informací a služeb. Bylo by proto vhodné, odlehčit zdravotníkům některé nadbytečné úkony a pacientům usnadnit přístup k aktuálním informacím, případně nabídnou funkce zjednodušující zdlouhavé procesy.

Cílem mého projektu je vytvořit webovou prezentaci Chrudimské nemocnice, a.s., která by měla návštěvníkům zprostředkovat snadný přístup k informacím a umožnit objednávání na vybraná pracoviště. Tím by se v budoucnu částečně odlehčila administrativa spojená s telefonickými objednávkami. Pacientům by tento systém měl přinést usnadnění v podobě snadno získatelných aktuálních informací o nemocnici a jejich pracovištích, na která se budou moci přímo objednat.

Představa klienta o projektu vychází ze stávající situace a komplikací k nimž dochází. Hlavními nedostatky původní webového sídla byly:

- Nepřehlednost,
- nemožnost úpravy textů bez placeného zásahu správce,
- omezené možnosti rozšíření,
- statická forma ukládání dat, která neumožňovala efektivně aktualizovat a pracovat s uchovávanými informacemi.

Nároky na nový systém tedy vychází z nedostatků předešlého řešení, doplněné o požadavky z praxe s ohledem i na použitelnost v budoucnosti. Nový systém by tedy měl být:

- Jednoduchý jak pro pacienty, kteří si budou prohlížet webové stránky, tak i pro zaměstnance, jež ho budou spravovat.
- Uchovávat údaje o pracovišti a jeho zaměstnancích. Z nich by měl čerpat objednávkový systém.
- Objednávkový systém umožní pacientovi registraci a poté vytvoření rezervace na určitém pracovišti a u konkrétního lékaře. Pacient by měl mít možnost zobrazit své objednávky na jednotlivých pracovištích.
- Všechny informace o pracovištích, zaměstnancích, ordinačních dobách, termínech pro objednání a dalších volbách by měly být v přehledné formě prezentovatelné a dostupné návštěvníkům stránek, kterým odpadne spousta starostí s jejich získáním.
- Systém by měl umožnit spravování veškerých údajů v administrátorském rozhraní, které by ovládal zaškolený pracovník s pokročilou znalostí práce na Internetu.
- Celý návrh má být řešen takovým způsobem, aby ze zadaných dat šla vygenerovat webová prezentace Nemocnice Chrudim, a. s. Generování má umožňovat jak tvorbu stránek, tak i samotného menu, které bude zpřístupňovat jednotlivé sekce. Je požadováno uchování hierarchické struktury pracovišť v databázi, které se využije pro jejich generování v menu.
- Velký důraz je kladen na možnost rozšíření celého systému bez nutnosti zásahů do zdrojového kódu. Při návrhu by se proto mělo dbát na univerzálnost a obecnost celého řešení.
- Provoz systému by neměl být příliš nákladný.

Pro další pokračování v projektu by bylo dobré vysvětlit, komu a jak má webová aplikace sloužit. Uživatel je chápán buď jako zaměstnanec s oprávněním, které mu dovolí přístup do určitých administračních sekcí systému, kde může přetvářet uložená data v databázi. Nebo bude vystupovat jako návštěvník, který vyhledává

nejrůznější informace, které jsou generovány z databáze, jejíž obsah je neustále aktualizován právě pověřenými pracovníky přes administrační rozhraní. K dispozici mu je také přehled volných termínů na pracovištích, kam se může po zaregistrování objednat.

Přístup k aplikaci zprostředkovává server, který generuje z požadavků uživatele souhrn informací, které mu pak zasílá přes bezstavový HTTP protokol. V případě potřeby zapamatování si předchozích stavů, jsou využity session. K uchování informací se využívá databáze MySQL. Na serveru musí také běžet php modul, pro podporu skriptů v programovacím jazyce php.

3.3 Způsoby zavedení objednávkového systému

K realizaci objednávkového systému pro Chrudimskou nemocnici, a.s. se nabízí několik možností. Nabídka je však do jisté míry omezena, protože objednávkové systémy ve zdravotnictví zatím nezaznamenaly patřičný rozvoj a hotových řešení není mnoho. Důvodem je především komplikovaný vývoj informačních technologií ve zdravotnictví, který nešel zrovna ideální cestou. Každá organizace používá libovolný informační objednávkový systém, data jsou v nejednotné formě, mnohdy i v podobě ručně psané dokumentace a pro vývojáře je těžké navrhnout řešení, které by vyhovovalo všem. Proto je vhodné si dané možnosti kategorizovat a dle nároků kladených na systém vybrat tu, která bude nejvíce vyhovovat.

Koupení hotového programového řešení

Tato sféra je zatím nejvíce oblíbená a rozšířená. Vyznačuje se svou jednoduchostí a poměrně příznivou cenou. Řešení v tomto sektoru představují většinou speciálně zaměřené informační systémy lokálního typu, kdy je vytvořen server, na který se připojují pomocí lokální sítě speciální klientské aplikace. Tyto umožňují měnit obsah informací v databázi. Systémy jsou většinou navrhovány tak, že vyhoví základním administrativním požadavkům zdravotníků.

Jeden z takovýchto produktů pochází od firmy Stapro, která se zabývá informačními technologiemi ve zdravotnictví. Systém pokrývá provoz klinických pracovišť řadou navzájem propojených produktů umožňujících vedení zdravotní dokumentace a podporujících provozní činnosti na jednotlivých pracovištích. Systém nabízí i objednávkový modul, který ale ovládá pouze zdravotní pracovník. Veškerá data jsou ukládána v lokální formě, která znemožňuje nasazení do online služeb, což je jeden z hlavních požadavků na zaváděný objednávkový systém.

Problém také nastává při větším počtu klientských stanic, kdy se takovéto řešení značně prodraží.

Nasazení bezplatné programové alternativy

Hlavním představitelem této kategorie jsou redakční systémy. Označují systém sloužící pro správu obsahu internetových prezentací. Celý systém se skládá z několika bloků, které jsou výstupem jednotlivých modulů. Každý z nich má specifické vlastnosti a možnosti a dohromady tvoří jeden celek. Jako přídatný modul lze také použít objednávkové systémy. Ty jsou však převážně zaměřeny na cestovní ruch a jeho odvětví. Použitím univerzálnějšího modelu, by se zase ztratila jednoduchost a rychlost jak na straně personálu, tak na straně zákazníka. Jedinou možností by v tomto případě bylo vyvinout speciální objednávkový modul pro zdravotnictví a napojit ho do redakčního systému. Výhodou tohoto řešení jsou nulové náklady na pořízení a užívání. Avšak vzhledem ke specifickým požadavkům na strukturu pracovišť, menu a mnoha dalších údajů, by mohlo být napojení na redakční systém značně problémové a omezující.

Pronájem služeb

Zajímavou alternativou je outsourcing objednávkového systému. Tato varianta s sebou přináší jednoduchost a bezproblémovost, vykoupená většinou vysokými provozními náklady. V České Republice je k dispozici začínající projekt s názvem MYSOLP. Tento online objednávkový systém zatím běží ve zkušebním režimu ve třech nemocnicích. Po zaregistrování, jak ze strany zdravotního zařízení, tak ze strany pacienta, je možno provádět objednávky na vybraných pracovištích a zvolit si prováděný úkon. Výhod má toto řešení hned několik. Pacient si může na jednom místě vybírat z velkého množství nabídek a zdravotní zařízení zase nabízí své služby mnohem větší skupině zákazníků než při jiných řešeních. V normálním provozu by pak měly být registrace, jak pro zdravotní pracoviště, tak i pro pacienty, zpoplatněny určitou paušální částkou na rok. Jelikož je ale tento systém teprve ve zkušební verzi a placení paušálních poplatků by se mohlo v budoucnu značně prodražit, není toto řešení nejvhodnější.

Vývoj projektu na míru

Toto řešení je výhodné v tom, že jeho funkcionalita vychází přesně ze zadaných požadavků, takže se dá očekávat při nasazení takto vytvořené aplikace nejvyšší možný výkon a komfort. Problém může být ve vyšších pořizovacích nákladech, větší

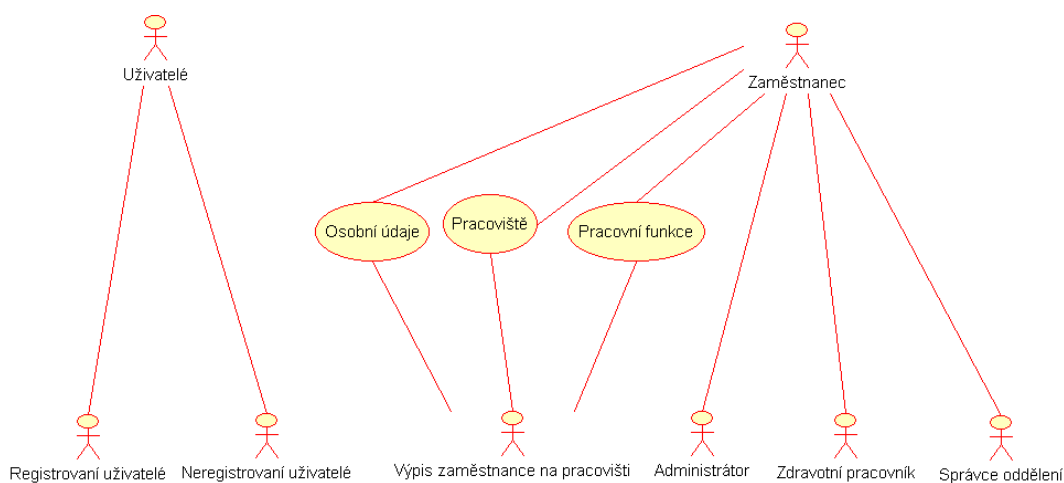
časové náročnosti vývoje, ve zvýšeném riziku výskytu problémů, jelikož aplikace nemohla být dostatečně testována v reálném provozu

Zhodnocení

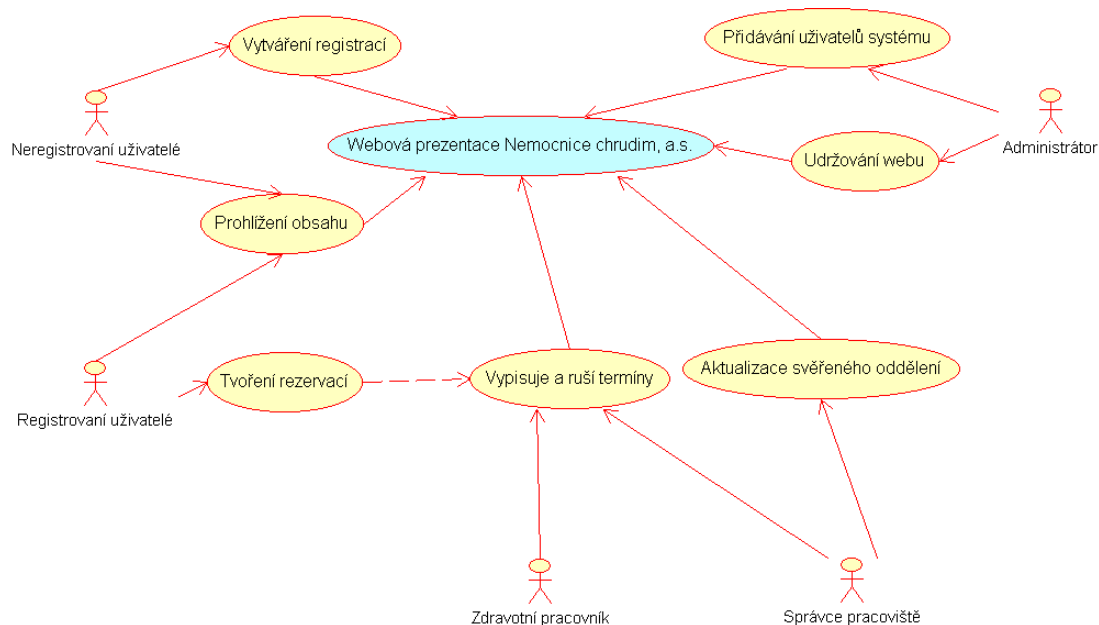
Každé ze zde uvedených řešení má své výhody a nelze jednoznačně říci, který systém je nejlepší. Proto se zhodnocení zaměřuje na použitelnost produktů vůči požadavkům kladených na objednávkový systém Chrudimské nemocnice, a.s. Dle těchto kritérií se jako nejvhodnější varianta jeví cesta *vývoje projektu na míru*, která zajistí specifické požadavky projektu bez jakýkoliv kompromisů. Její negativa jsou ze strany klienta akceptovatelná, takže nic nebrání k použití této varianty.

3.4 Analýza objednávkového systému

Zobrazuje chování systému z hlediska uživatele. Na níže uvedených obrázcích jsou znázorněny typy uživatelů a jejich role v systému. V celém systému vystupují různí uživatelé s odlišnými úrovněmi oprávnění, podle kterých jim je umožňován přístup do jednotlivých sekcí stránek.



Obr. 2: Hierarchické zobrazení aktérů systému



Obr. 3: Diagram užití systému

Systém bude rozpoznávat šest typů uživatelů, kteří mají tyto funkce:

- **Neregistrovaní uživatelé** – se volně pohybují na stránkách bez nutnosti přihlášení. V případě, že chtějí uskutečnit objednávku, se musí přihlásit do systému, čemuž předchází vytvoření registrace.
- **Registrovaní uživatelé** – mají po přihlášení přístup ke svému účtu, kde mohou uskutečňovat rezervace na jednotlivá pracoviště a sledovat své objednávky.
- **Výpis zaměstnance na pracovišti** – zaměstnanec v systému může také představovat pasivního uživatele, který se nijak na přetváření webu nepodílí, ale slouží pouze jako zdroj dat pro výpis kontaktních údajů zaměstnance, který na daném pracovišti plní určitou funkci.
- **Administrátor** – disponuje nejvyššími právy, která mu umožní spravovat kompletně celý web bez omezení.
- **Zdravotní pracovník** – reprezentuje zaměstnance, který se stará o objednávání pacientů na svém oddělení. Kontroluje počet objednaných pacientů na určité datum, ruší termíny pro objednání nebo přímo jednotlivé objednávky.
- **Správce oddělení** – K oprávněním zdravotního pracovníka, jakými jsou rušení objednávek a rušení termínů pro objednání, přibírá navíc možnost editace dat spojených s pracovištěm, pod které spadá. Takže může měnit veškeré informace, které si návštěvník o daném pracovišti zobrazí.

Scénáře případů užití

Scénáře obsahují sekvenci událostí pro jednotlivé typy uživatelů systému. Mají nastínit předpokládané užití systému. U všech zaměstnaneckých scénářů je předpokládán proces přihlášení do systému a po dokončení operace proces odhlášení ze systému.

Scénář administrátora

1. Vytvoř zaměstnance.
 1. Zadej osobní údaje o zaměstnanci.
 2. Vytvoř zaměstnance.
2. Vytvoř pracovní funkci pro zaměstnance.
 1. Zadej název nové funkce.
 2. Vytvoř pracovní funkci.
3. Vytvoř typ pracoviště.
 1. Zadej název typu pracoviště.
 2. Vytvoř typ pracoviště.
4. Vytvoř pracoviště.
 1. Vyber typ pracoviště.
 2. Vyber nadřazené pracoviště.
 3. Zadej název pracoviště.
 4. Zadej doplňující údaje o pracovišti.
 5. Vytvoř pracoviště.
5. Přidej zaměstnance na vybrané pracoviště.
 1. Vyber pracoviště.
 2. Vyber pracovní funkci.
 3. Vyber zaměstnance.
 4. Proveď přidání.
6. Vytvoř ordinační dobu.
 1. Vyber pracoviště.
 2. Vyber zaměstnance.
 3. Nastav časové údaje.
 4. Vytvoř ordinační dobu.
7. Zrušení termínů pro objednání .
 1. Vyber ordinační dobu prezentovanou časem, ve který se má začít ordinovat, příjmením zaměstnance a dnem v týdnu ordinační doby.

2. Nastav datum, ve kterém se nebude ordinovat.
3. Proveď zrušení termínu pro objednání.
8. Vytvoř typ položky menu.
 1. Zadej název typu položky menu.
 2. Vytvoř typ položky menu.
9. Vytvoř vlastnosti položky menu.
 1. Vyber typ položky menu.
 2. Zadej hodnotu vlastnosti.
 3. Zadej název vlastnosti.
10. Vytvoř menu
 1. Zadej potřebné údaje pro vytvoření položky menu.
 2. Generuj podsložky k dané položce menu.
11. Výpis objednávek.
 1. Vyber pracoviště.
 2. Vypiš objednávky pro vybrané pracoviště.
 3. Zruš vybranou objednávku.
12. Vytvoř přihlašovací údaje.
 1. Vyber podle osobních údajů uživatele, kterému bude přiřazeno oprávnění.
 2. Zadej login, pod kterým se bude nový uživatel přihlašovat.
 3. Zadej heslo, pod kterým se uživatel bude přihlašovat.
 4. Zadej úroveň oprávnění.
 5. Dokonči vytvoření přihlašovacích údajů.

Scénář správce pracoviště

1. Editace přihlašovacích údajů správce.
 1. Zadej původní heslo.
 2. Zadej nové heslo.
2. Editace informací o pracovišti.
 1. Změň libovolný údaj o pracovišti.
 2. Ulož změnu.
3. Vypiš všechny objednávky na pracovišti.
 1. Zruš vybranou objednávku.
4. Zrušení termínů pro objednání.

1. Vyber ordinační dobu prezentovanou časem, ve který se má začít ordinovat, příjmením zaměstnance a dnem v týdnu ordinační doby.
2. Nastav datum, ve kterém se nebude ordinovat.
3. Proveď zrušení termínu pro objednání.

Scénář zdravotního pracovníka

1. Vypiš všechny objednávky na pracovišti.
2. Zruš vybranou objednávku.
3. Zrušení termínů pro objednání.
 1. Vyber ordinační dobu prezentovanou časem, ve který se má začít ordinovat, příjmením zaměstnance a dnem v týdnu ordinační doby.
 2. Nastav datum, ve kterém se nebude ordinovat.
 3. Proveď zrušení termínu pro objednání.

Scénář neregistrovaného uživatele

1. Prohlížení obsahu webové prezentace.
2. Zjišťování volných termínů pro online objednání.
3. Registrace do systému.

Scénář registrovaného uživatele

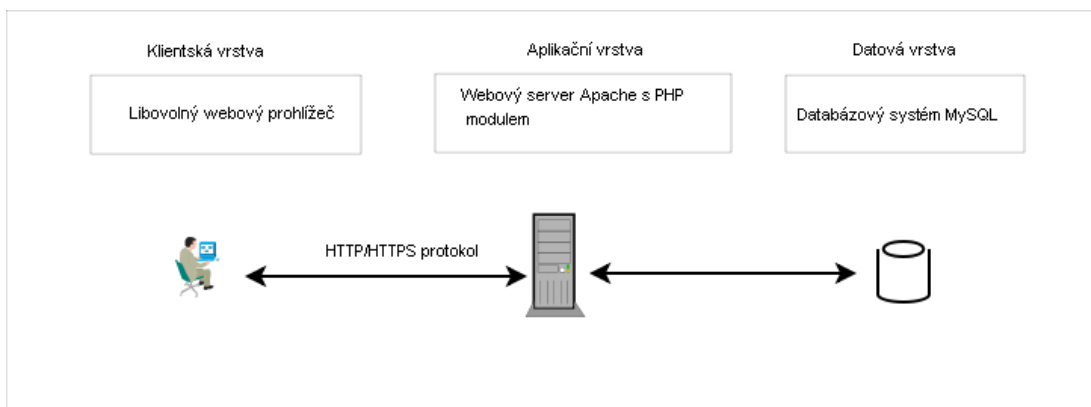
1. Prohlížení obsahu webové prezentace.
2. Zjišťování volných termínů pro online objednání.
3. Přihlášení do systému.
4. Výpis uskutečněných objednávek.
5. Editace uživatelského profilu.

3.5 Architektura objednávkového systému

Architektura systému je odvozena z třívrstvého modelu skládajícího se z klientské, aplikační a databázové části. Byla zvolena především pro svou výkonnost při vyšším počtu připojených uživatelů a nenáročnost v požadavcích na klientské vybavení.

Do systému není možný jiný přístup, než pomocí webového prohlížeče. Bez ohledu na to, zda se připojuje pacient ze zahraničí, nebo zaměstnanec provádějící různé datové operace z vlastní kanceláře. Vždy je přístupováno pomocí webového prohlížeče. Základní výhoda tohoto přístupu spočívá ve velmi snadné správě a údržbě celého systému. Na straně koncového klienta je nutná pouze běžná podpora

pro standardní prohlížeč, která nepřináší žádné speciální požadavky ani další náklady. V prostředí nemocniční prezentace se předpokládá návštěvnost v desítkách až stovkách klientů za den. Jakákoliv změna v aplikaci je pak propagována všem a ihned, což přináší v podstatě největší aktuálnost prezentovaných informací. Druhou výhodou je samozřejmě přístup z libovolného místa připojeného do Internetu, kterou mohou využít nejen pacienti, ale i pracovníci, kteří se systémem pracují například z domova nebo na cestách. Kvůli zvýšeným nárokům na zabezpečení aplikace bude použit do zabezpečených částí šifrovaný HTTP protokol, tedy HTTPS.

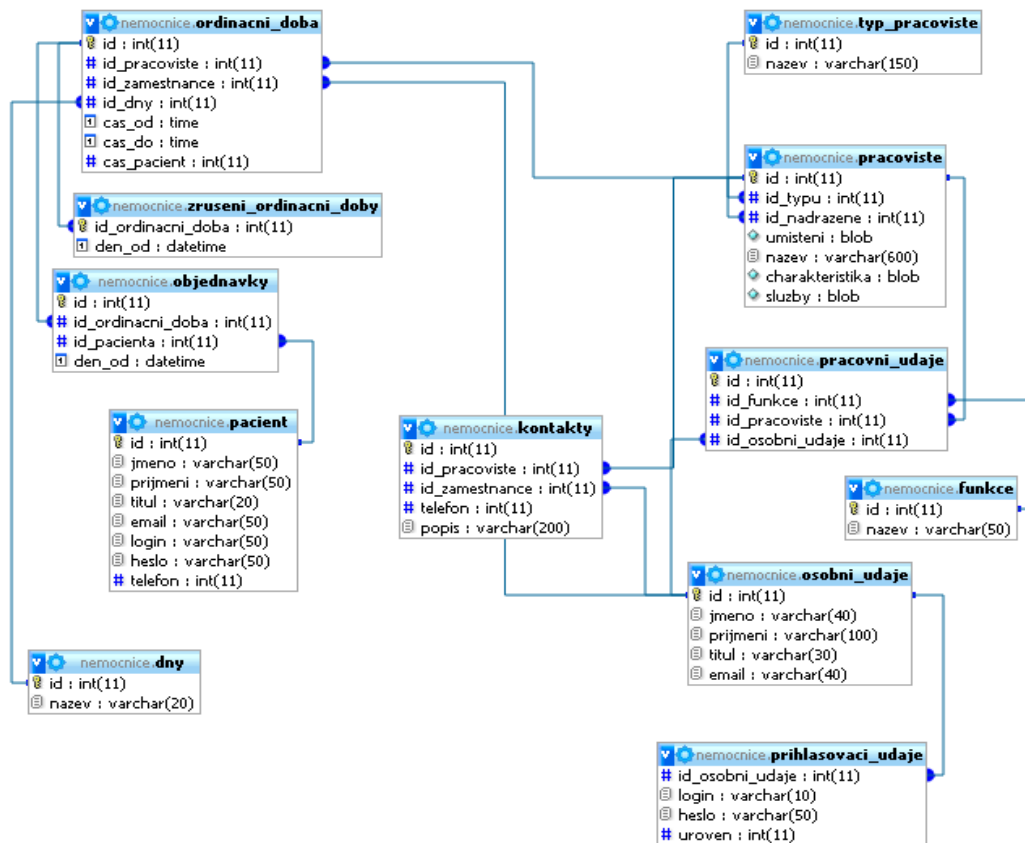


Obr. 4: Architektura systému

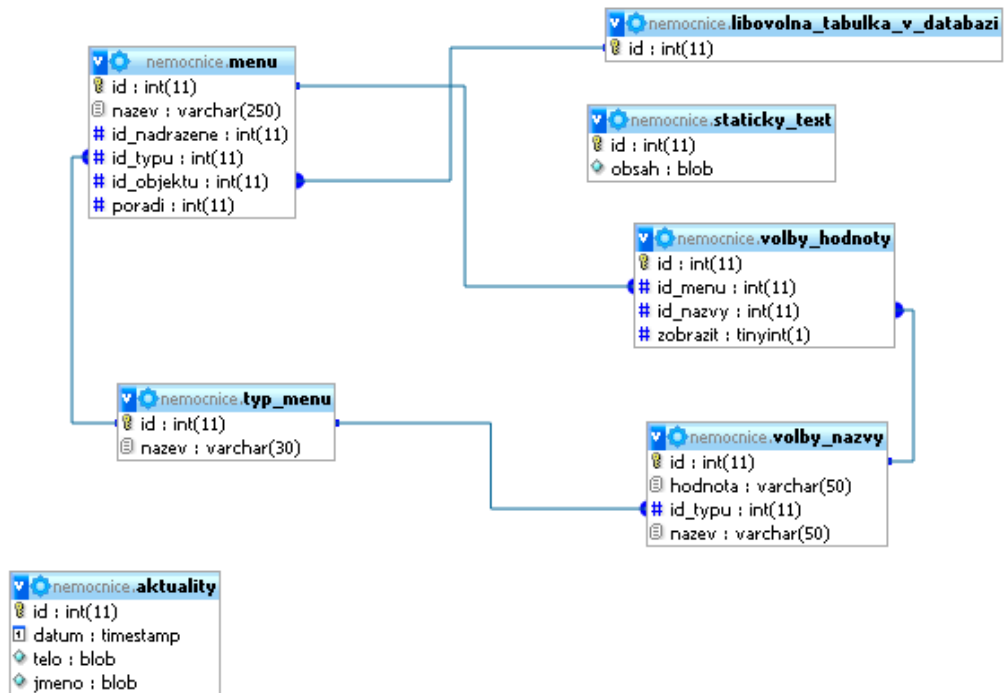
Hlavním úkolem systému je tedy umožnit pověřeným zaměstnancům sestavit ze zadaných informací webové sídlo Chrudimské nemocnice, a.s., které bude mimo jiné umožňovat návštěvníkům online objednání na vybraná pracoviště a jednotlivým zdravotníkům dle jejich oprávnění potom jednoduchou správu dat v systému.[6]

3.6 Návrh databáze

Databáze byla navrhována s ohledem na dodržení prvních třech normálních forem, tedy 1NF, 2NF, 3NF. V návrhu je použita fiktivní tabulka s názvem *libovolna_tabulka_v_databazi*, která nyní zastupuje tabulky: *pracoviste*, *typ_pracoviste* a *statický text*. Je zde proto, aby naznačila, že atribut *id_objektu* z tabulky *menu* určuje index libovolné tabulky v databázi, jejíž název je uložen v tab. *typ_menu*. Smyslem tohoto kroku je vytvořit v administraci možnost volby pro generování obsahu položky menu z jakékoliv tabulky. Pak je tedy možné zobrazit návštěvníkovi pomocí menu jakoukoliv informaci v databázi i s ohledem na pozdější rozšíření bez nutnosti zásahu do zdrojového kódu.



Obr. 5. ERD diagram první část



Obr. 6: ERD diagram druhá část

Přehled a popis tabulek

Zobrazuje informace o tabulkách použitých v objednávkovém systému. Důležité tabulky jsou vypsány podrobně a zbylé pouze s popisem, jakou mají funkci. Označení PK ve sloupci Klíč je zkratkou pro primární klíč a označení FK zase pro cizí klíč.

Základní popis tabulek:

- *Aktuality*, slouží k uchování aktualit.
- *Dny*, obsahuje id dnů v týdnu a jejich názvy. Nutné pro výpis kalendáře.
- *Funkce*, obsahuje názvy pracovních funkcí, např. doktor.
- *Kontakty*, umožňují uchovat dvě telefonní čísla pro zaměstnance na pracovišti. V případě, že má zaměstnanec víc pracovišť, je u každého pracoviště evidován i s jiným telefonním číslem.
- *Menu*, uchovává informace o hlavním menu.
- *Objednavky*, obsahují informace o uskutečněných rezervacích.
- *Ordinacni doba*, nese informace, ze kterých se vypisují nebo ruší termíny pro objednání a generuje tabulka ordinačních dob.
- *Osobni udaje*, obsahují základní data o zaměstnancích.
- *Pacient*, obsahuje souhrn nezbytných informací o pacientovi.

- *Pracoviste*, je tabulka pro univerzální objekt, který může reprezentovat jakýkoliv pracovní útvar. Detailní popis v tab. 1.
- *Pracovni udaje*, slouží k propojení informací o pracovníkově osobních údajích a pracovišti, na kterém vykonává určitou pracovní funkci. Detailní popis v tab. 2.
- *Sticky text* slouží k uchování informací z WYSIWIG editoru, které je možné přidávat jako alternativní obsah u položek menu.
- *Typ menu*, obsahuje názvy tabulek, ze kterých se budou vybírat informace pro reprezentaci položky v menu. Detailní popis v tab. 3.
- *Typ pracoviste* doplňuje údaje k tabulce pracoviště.
- *Volby_hodnoty*, uchovávají informaci o tom, které vlastnosti u generované položky menu se budou vypisovat. Detailní popis v tab. 5.
- *Volby_nazvy*, definují jak se budou vlastnosti v tabulce volby_hodnoty nazývat a k jakému typu menu daná vlastnost patří. Detailní popis v tab. 4.
- *Zruseni_ordinacni_doby*, obsahuje termíny ordinačních dob, pro které se nebudou vypisovat objednávky.

Detailní popis stěžejních tabulek:

Klíč	Název sloupce	Datový typ	Poznámka
PK	id	Integer	
FK	id_typu	Integer	Určuje typ oddělení, cizí klíč do tab. typ_oddeleni(id).
FK	id_nadrazene	Integer	Nadřazené pracoviště, cizí klíč do tab. pracoviste(id).
	umisteni	Blob	Informace o umístění pracoviště.
	nazev	Varchar(600)	Pojmenování pracoviště.
	charakteristika	Blob	Informace charakterizující funkci pracoviště.
	sluzby	Blob	Přehled služeb prováděných na pracovišti.

Tab. 1: Detailní popis tabulky pracoviste

Klíč	Název sloupce	Datový typ	Poznámka
PK	id	Integer	
FK	id_funkce	Integer	Určuje pracovní funkci zaměstnance, cizí klíč do tab. funkce(id).
FK	id_pracoviste	Integer	Váže údaj ke konkrétnímu pracovišti, cizí klíč do tab. pracoviste(id).
FK	id_osobni_udaje	Blob	Váže údaj ke konkrétnímu zaměstnanci, cizí klíč do tab. osobni_udaje(id).

Tab. 2: Detailní popis tabulky pracovní udaje

Klíč	Název sloupce	Datový typ	Poznámka
PK	id	Integer	
FK	id_funkce	Integer	Určuje pracovní funkci zaměstnance, cizí klíč do tab. funkce(id).
FK	id_pracoviste	Integer	Váže údaj ke konkrétnímu pracovišti, cizí klíč do tab. pracoviste(id).
FK	id_osobni_udaje	Blob	Váže údaj ke konkrétnímu zaměstnanci, cizí klíč do tab. osobni_udaje(id).

Tab. 3: Detailní popis tabulky menu

Klíč	Název sloupce	Datový typ	Poznámka
PK	id	Integer	
FK	hodnota	Varchar(50)	Reprezentuje, jak se bude jmenovat daná vlastnost.
FK	id_typu	Integer	Určuje pro jaký typ je vlastnost položky v menu dostupná, cizí klíč do tab. typ_menu(id).
FK	nazev	Varchar(50)	Určuje název vlastnosti, který se bude zobrazovat administrátorovi.

Tab. 4: Detailní popis tabulky volby_nazvy

Klíč	Název sloupce	Datový typ	Poznámka
PK	id	Integer	
FK	id_menu	Integer	Připojuje k položce v menu určitou vlastnost, cizí klíč do tab. menu(id).
FK	id_nazvy	Integer	Slouží k přístupu k detailním informacím připojené vlastnosti položky, cizí klíč do tab. volby_nazvy(id).
FK	zobrazit	Bool	Říká, zda se daná vlastnost bude vypisovat po kliknutí na položku či nikoliv.

Tab. 5: Detailní popis tabulky volby_hodnoty

4. Implementace objednávkového systému

Kapitola zachycuje implementaci webové prezentace, její strukturu a řešení zabezpečení celé aplikace. Jsou zde vyznačeny nejproblémovější části aplikace a nastíněno jejich řešení. V závěru je popsán návod pro všechny uživatelské úrovně systému.

4.1 Adresářová struktura zdrojových souborů

Pokud bude potřeba modifikovat nějakou část webové aplikace, je nutné znát, kde jsou umístěny soubory vyžadující úpravu. V této sekci je popsána adresářová struktura zdrojových souborů v hlavním adresáři nemocnice.

Následující adresáře se nacházejí v kořenovém adresáři:

- **administrace** – Obsahuje soubory a adresáře spojené s administrační částí webové aplikace.
- **aktuality** – Pouze dočasné řešení administrace aktualit, která bude přesunuta do administrační části, takže tento adresář zmizí.
- **images** – Obsahuje obrázky použité pro vytvoření celkového designu aplikace.
- **kalendar** – Je adresář pro open source komponentu, která zajišťuje zobrazení kalendáře při výběru data.

Pro instalaci aplikace jsou nutné ještě další informace, které obsahuje programátorská dokumentace v příloze E.

4.2 Knihovny funkcí

Celé aplikace je tvořena pomocí php skriptů, které jsou členěny podle použití do jednotlivých funkcí a ty jsou pak řazeny do knihoven. Pro větší bezpečnost a přehlednost jsou knihovny používány v systému rozdělené takto:

- `fce_pro_vsechny.php` – knihovna obsahuje funkce, které jsou dostupné pro všechny uživatele systému. Jsou především zaměřeny na problematiku pacienta.

- fce_admin.php – je knihovna funkcí sloužící výhradně pro zaměstnance. Poskytuje nástroje pro rozsáhlou administraci webového rozhraní.

Ukázky problémových částí

Při psaní kódu jsem často narážel na nejrůznější problémy, jejichž řešení mnohdy bylo jednoduché, ale trvalo poměrně dlouho, než jsem na něj přišel. Proto jsou zde tyto problematické pasáže popsány a nastíněn způsob jejich řešení.

Menu

Poměrně triviální záležitost, vytvoření hierarchického menu se v praxi stala docela složitou. Při výpisu totiž bylo třeba brát zřetel na posloupnost vypisujících se položek, aby bylo zanoření správně zobrazeno. A postupným nabalováním dalších potřebných prvků vznikla tato rozsáhlá funkce, umožňující výpis všech položek menu s oddělenou první úrovní, která má speciální formátování oproti dalším úrovním.

```
function vypis_potomka_uzivatel($co,$id_rodice){
    spojeni();//Funkce pro připojení do databáze.
    //Pro výpis aipky, která bude jen na první úrovni menu, kdy~ je je
    id_rodice==0.
    $sql_sipka=mysql_query("SELECT id_nadrazene FROM $co WHERE
    id='$id_rodice'");
    $data_sipka=mysql_fetch_object($sql_sipka);
    $id_prarodice=$data_sipka->id_nadrazene;
    $sql=mysql_query("SELECT id FROM $co WHERE id_nadrazene='$id_rodice'
    GROUP BY poradi");
    $radku=mysql_num_rows($sql);
    if($radku== 0){ return (0) ;} //}ádný potomek není,vrátíme id
    a ukončíme funkci.
    else{//Jeden a více potomků
    //Zjištění názvu pro výpis první řady.
    $sql_nazev=mysql_query("SELECT nazev from menu where
    id='$id_rodice'");
    $data_nazev=mysql_fetch_object($sql_nazev);
    if($id_rodice!=0) //Výpis rodiče.
    //Podmínka pro vypsání aipky.
    if($id_prarodice==0){$_SESSION[menu]=$_SESSION[menu]. '<li
    class="sipka"><a href="index.php?id_menu='.$id_rodice.'"> '.
    $data_nazev->nazev.' </a><ul>';}
    else{$_SESSION[menu]=$_SESSION[menu]. '<li ><a href="index.php?
    id_menu='.$id_rodice.'"> '.$data_nazev->nazev.' </a><ul>';}
    //Pouze jeden potomek.
    if($radku==1){ $data=mysql_fetch_object($sql);
    $sql_nazev=mysql_query("SELECT nazev from menu where id='$data-
    >id'");
    $data_nazev=mysql_fetch_object($sql_nazev);
    //Pokud tento jediný potomek nemá potomka, pak ho vypis a ukonči
    funkci, jinak se zase zanoř a vyhledej další potomky.
    if (vypis_potomka_uzivatel($co,$data->id)==0 ) {
    //Výpis aipky, která bude jen na první úrovni, kdy~ je je
    id_rodice==0.
    if($id_rodice==0){$_SESSION[menu]=$_SESSION[menu]. '<li
    class="sipka"><a href="index.php?id_menu='.$data->id.'"> '.
    $data_nazev->nazev.' </a></li>';}
```

```

else{$_SESSION[menu]=$_SESSION[menu]. '<li><a href="index.php?
id_menu='.$data->id.'"> '.$data_nazev->nazev.' </a></li>'; } }}
    else{ while($data=mysql_fetch_object($sql)){//Výpis dvou
a více potomků
if (vypis_potomka_uzivatel($co,$data->id)==0 ){//Pokud nemá žádné
děti, vypis obsah.
$id=$data->id;
$sql_nazev=mysql_query("SELECT nazev from menu where id='$id'");
$data=mysql_fetch_object($sql_nazev);
//Výpis aipky, která bude jen na první úrovni, kdy~ je je
id_rodice==0.
if($id_rodice==0){
$_SESSION[menu]=$_SESSION[menu]. '<li
class="sipka"><a href="index.php?id_menu='.$id.'"> '.$data-
>nazev.' </a></li>';
}else{$_SESSION[menu]=$_SESSION[menu]. '<li><a href="index.php?
id_menu='.$id.'"> '.$data->nazev.' </a></li>';}
}}}
    if($id_rodice!=0)$SESSION[menu]=$SESSION[menu].
'</ul></li>';
return(1);
}

```

Celkový čas na pacienta

Jeden velký problém byla práce s časem. Jelikož v php chybí celá řada časových funkcí, je nutné je neustále složitě převádět z jednoho formátu na druhý a pak vznikají poměrně složité dotazy na přitom tak jednoduché požadavky. Cílem níže uvedeného dotazu bylo odečíst od sebe dva časové údaje a získat jejich rozdíl v minutách, který se pak zaokrouhlí. Podle něho se poté do výpisů bude připočítávat čas na jednoho pacienta.

```

$sql_rozdil=mysql_query("SELECT MINUTE((SELECT TIMEDIFF((SELECT
cas_do FROM ordinacni_doba WHERE id='$data_terminy->id'),(SELECT
cas_od FROM ordinacni_doba WHERE id='$data_terminy->id' ))) AS
mi, HOUR((SELECT TIMEDIFF((SELECT cas_do FROM ordinacni_doba WHERE
id='$data_terminy->id'),(SELECT cas_od FROM ordinacni_doba WHERE
id='$data_terminy->id' ))) AS ho;");

```

Výpis tagu SELECT

Při neustále se opakující tvorbě formulářů a jejich prvků jsem se snažil práci si co nejvíce zefektivnit a vyhnout se opakovanému psaní jedné části kódu pořád dokola. Proto vznikla tato funkce na vypsání a naplnění HTML tagu SELECT. Funkce počítá i s nastavením výchozí hodnoty, na kterou má být SELECT nastaven. Toho pak využívám při editaci položek k zapamatování nastavených hodnot. Do položky SELECTU je možné zapsat až tři sloupce z tabulky.

```

function vypis_select($vystup,$id,$sql_vypis_vse,$sql_nazev,
$popisek,$nazev_sql,$nazev_sql2,$nazev_sql3){
echo '      <label class="admintext" for="'.$popisek.'">'.
$popisek.'</label><br />
      <select id="'.$popisek.'" name="'.$vystup.'">          ';
//-1 Znamená, ~e polo~ka bude nastavena na mo~nost ==~ádný výběr==

```

```

        if ($id==-1){echo ' <option value="-1"
selected="selected">==~ádný výběr==</option>' ;}
        else {
            $data=mysql_fetch_object($sql_nazev);
            echo '<option value="'. $id.'"
selected="selected">'. $data->$nazev_sql.' '. $data->$nazev_sql2.' '.
$data->$nazev_sql3.' </option>';
            echo '<option value="-1" >==~ádný
výběr==</option>' ;
        }
        while($zaznam=mysql_fetch_object($sql_vypis_vse))
        {
            if($zaznam->id!=$id){echo '<option value="'.
$zaznam->id.'">'. $zaznam->$nazev_sql.' '. $zaznam->$nazev_sql2.' '.
$zaznam->$nazev_sql3.'</option>' ; }//podminka,aby se nevypsala uz
vypsana polozka
        }
        echo '</select><br />';
    }
}

```

Použití funkce *vypis_select*:

```

//$vystup ...podle něho se bude jmenovat vystupní proměnná selectu
//id Konkrétní id, pokud je zadáno nastavi se na jeho hodnotu,jinak
na -1
//Nastavení potřebných údajů
$sql_nazev="SELECT id,nazev FROM typ_menu WHERE id='$dotaz->id_typu'
";
$sql_vypis_vse="SELECT id,nazev FROM typ_menu GROUP BY id ";
$sql_vypis_vse=mysql_query($sql_vypis_vse);
$sql_nazev=mysql_query($sql_nazev);
//Použití funkce
vypis_select($vystup="id_typu",$id=$dotaz->id_typu,$sql_vypis_vse,
$sql_nazev,$popisek="Typ menu",$nazev_sql="nazev"," "," ");

```

Generování menu

Pro odlehčení serveru a pro rychlejší odezvu na straně klienta, je po prvním načtení webové prezentace uloženo vygenerované menu do session proměnné, ze které se při každém dalším požadavku na menu jen načte. Zde je část kódu, která zajišťuje zobrazení vygenerovaného menu .

```

<?php
if(!isset($_SESSION['menu'])) {
Spojeni();//Funkce pro připojení do databáze
$vysledek=mysql_query("select id, jmeno from aktuality ORDER BY
id");
$radku=mysql_num_rows($vysledek);
if ($radku==0) {
    echo "}ádné položky v menu";
} else//Výpis menu
{
    $_SESSION[menu]= '<div id="menu"><ul>';
    vypis_potomka_uzivatel("menu",0);
    $_SESSION[menu]=''. $_SESSION[menu]. '</ul></div>';
    echo $_SESSION[menu];
}
}

```



```
else{echo $_SESSION[menu];}  
?>
```

CSS stylování

Pro rychlejší načítání základního vzhledu, byly opakující se části pozadí rozřezány na malé jedno pixelové obdélníky, které se kopírovaly buď ve vodorovném nebo svislém směru. Ukázka zdrojového kódu jedné z definic ze souboru style.css

```
.menu{  
position: absolute;  
top: 10px;  
left: 7px;  
width: 196px;  
background-image: url(images/menu_bg.jpg);  
background-repeat: repeat-y;  
z-index:2;  
}
```

4.3 Bezpečnost aplikace

Zabezpečení se týká autentizace a poté následné autorizace uživatele do aplikace. Při autorizaci do administrátorské sekce se vyhodnotí úroveň oprávnění, podle které jsou zobrazovány jednotlivé nabídky v menu a omezovány volby vybraných funkcí. Tato kontrola se provádí na základě členství uživatele v různých uživatelských skupinách – administrátor, správce pracoviště, zdravotní pracovník.

Přihlašování pacientů do systému funguje na stejném principu, ale je od zaměstnaneckého systému odděleno použitím jiné tabulky pro ověřování údajů a k dispozici má jiné funkce, přizpůsobené pro pacienty.

Hesla se kvůli větší bezpečnosti do databáze neukládají, ale jsou uloženy pouze jejich hashe, které generuje hashovací funkce SHA1. Při práci s hesly se vždy využívá výsledný hash a samotné heslo je tak chráněno proti zneužití.

Všechny formuláře a vstupní data, jsou ošetřena proti SQL injection. Což znamená, že útočník nemůže podvrhnout vstupní data tak, aby byl nějakým způsobem pozměněn výsledek SQL dotazu. Nejčastější případy útoků probíhají přes neošetřené formuláře na webových stránkách.

V tomto skriptu probíhá autentizace přihlašovaného uživatele do systému. V případě úspěchu je pak autorizován.

```
<?php  
session_start();  
include "../fce_admin.php";  
spojeni_admin(); // připojení do databáze
```

```

if (isset($_POST['odhlaseni'])) { //odhlášení ze systému
unset($_SESSION['login'],$_SESSION['id_osobni_udaje'],$_SESSION['uroven']);
mysql_close(); //ukonceni spojeni s databazi /1/
}else {
$login = mysql_real_escape_string($login);
$heslo = SHA1($heslo);
$dotaz = "SELECT login,heslo,id_osobni_udaje,uroven FROM prihlasovaci_udaje
WHERE login='$login' AND heslo='$heslo'";
$sql = mysql_query($dotaz);
$data=mysql_fetch_object($sql);
$pocet = mysql_num_rows($sql);
    if($pocet == "1") {
        $_SESSION['login'] = stripslashes($data->login);
        $_SESSION['id_osobni_udaje'] =$data->id_osobni_udaje;
        $_SESSION['uroven'] =stripslashes($data->uroven);
        mysql_close(); //ukonceni spojeni s databazi /1/
    }else{
        $_SESSION['uroven']=0;
        mysql_close(); //ukonceni spojeni s databazi /1/
    }
}
}
header("Location:index.php");
exit;
?>

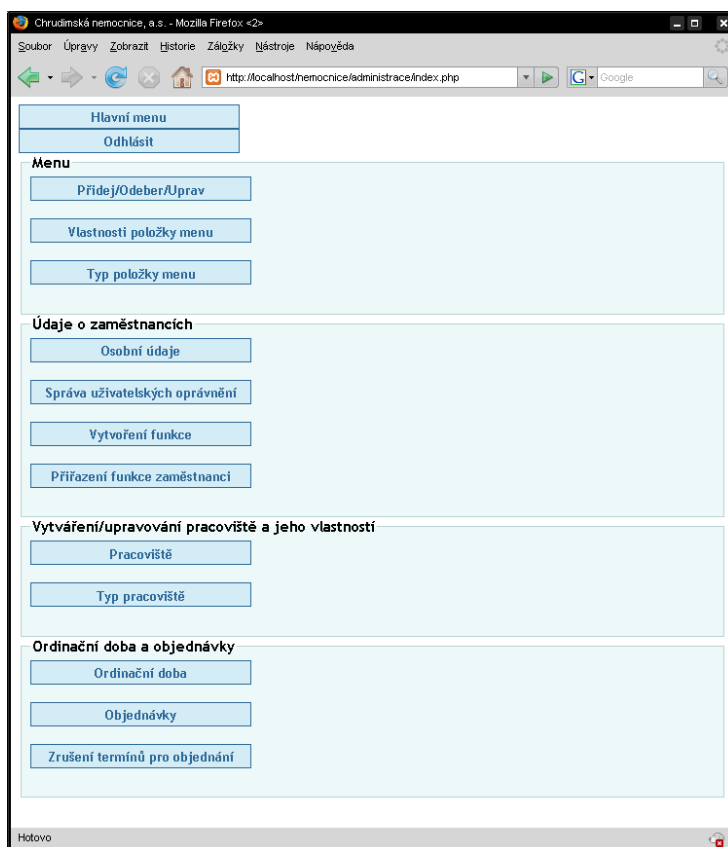
```

Posledním krokem k zabezpečení aplikace je použití zabezpečeného protokolu HTTPS. Tento protokol vychází z původního HTTP s tím rozdílem, že před odesláním jsou data zašifrována a elektronicky podepsána. Šifrování zabraňuje odposlechu komunikace mezi zúčastněnými stranami. Elektronický podpis zase ověří pravost druhé strany, se kterou se snažíme komunikovat, a tím zabrání tomu, aby se za ni někdo neoprávněně vydával. Je proto tedy bezpodmínečně nutné při práci s citlivými daty použít zabezpečený přenos, a tím maximalizovat jejich ochranu. Na druhou stranu sekce, které nevyžadují zabezpečení je zbytečné šifrovat HTTPS protokolem, jelikož výsledná komunikace by se pak značně zpomalila v důsledku nemalé režie dat, která vzniká při jejich šifrování.

4.4 Základní funkcionality aplikace

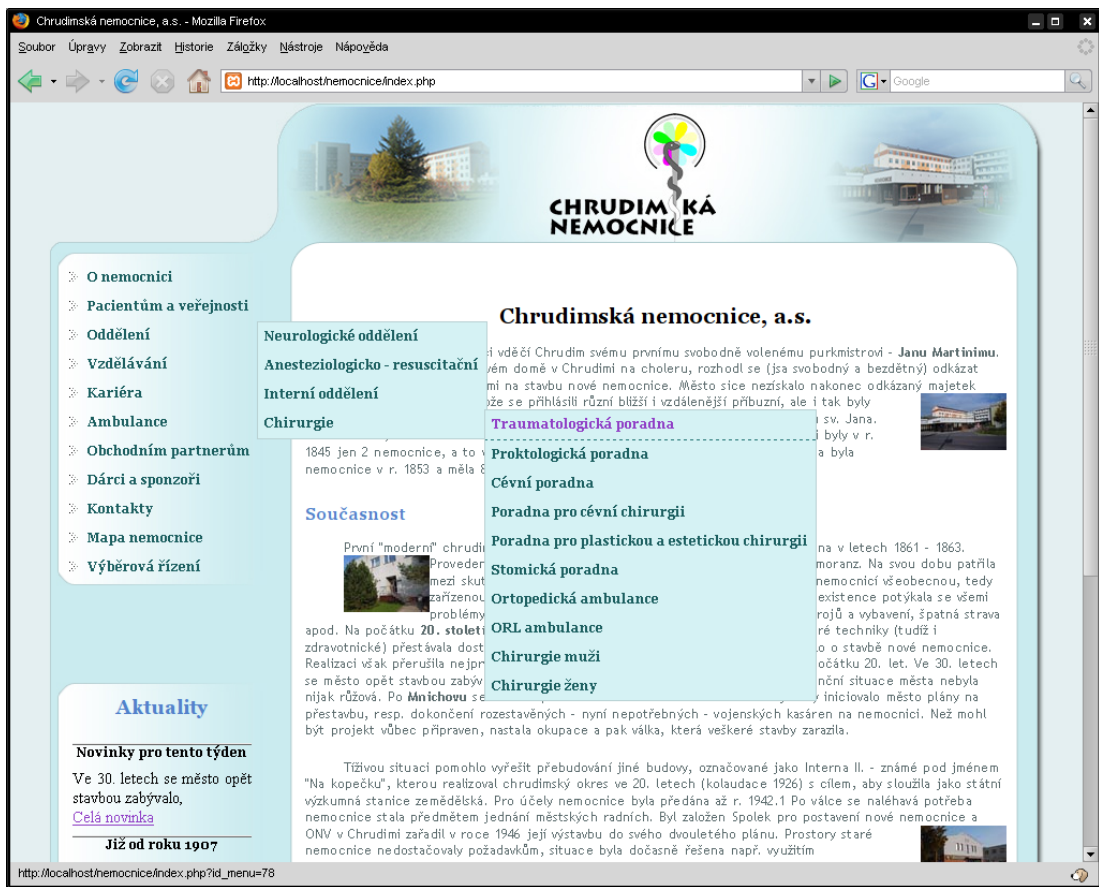
Aplikace se dělí na dvě části, první tzv. Front-End uvedený na obr. 7, slouží zaměstnancům, kterým umožňuje manipulovat s uloženými daty a přetvářet jejich obsah, ten se pak bude zobrazovat návštěvníkům webu. Systém pro tento účel rozlišuje tři úrovně uživatelů. Nejnižší úroveň je zdravotní pracovník, který je omezen pouze na správu objednávek na svém pracovišti. Detailní popis jeho voleb je uveden v příloze C. Následující úroveň správce pracoviště rozšiřuje možnosti úprav

na celé pracoviště, které je danému správci přiděleno. Může například měnit informace o pracovišti nebo přidávat ordinační doby. Podrobněji o volbách správce pracoviště v příloze B. Nejvyšší a neomezenou úroveň reprezentuje administrátor, který má přístup ke všem volbám bez omezení. Může například vytvářet nové pracoviště nebo také zaměstnance, kterým pak přidá pracovní funkce a oprávnění. Určuje, jaké informace a v jaké struktuře se budou návštěvníkům zobrazovat. Detailní popis celého administrátorského rozhraní je v příloze A.



Obr. 7: Administrátorské rozhraní

Druhou část, tzv. Back-End, k vidění na obr. 8, využívají řadoví uživatelé Internetu, kterým jsou generovány informace zadané zaměstnanci. Kromě standardních funkcí, které souvisí se zobrazováním obsahu, mají možnost i online objednání na pracoviště, které má vypsané volné termíny.



Obr. 8: Vnější vzhled webové prezentace.

5. Závěr

Tato práce se zaměřuje na problematiku objednávkového systému v nemocničním prostředí. Jeho nasazením se může ušetřit mnoho času a nákladů oběma zúčastněným stranám v procesu objednávání. Systém by měl plnit funkci klasických webových stránek s objednávkovým modulem, který by v počátcích spíše doplňoval stávající telefonické objednávání. V budoucnu by se situace měla změnit ve prospěch online objednávání.

Vzhledem k rozsáhlosti celého projektu, nebylo možné dokončit některé jeho části, jež by dotvořily komplexnost celého systému. Avšak základní kostra je plně funkční a splňuje požadavky, které byly v úvodu zadány. Tedy možnost udržovat celý web pomocí administrátorského rozhraní, ze kterého se po vložení všech potřebných dat snadno vygeneruje webové sídlo Chrudimské nemocnice, a.s. Součástí systému je také objednávkový modul, umožňující rezervaci pacienta k vybranému lékaři. Uživatelská část aplikace je optimalizována na rychlost, přehlednost a kompatibilitu s moderními webovými prohlížeči.

Před nasazením aplikace do ostrého provozu by však bylo třeba nejprve vyřešit tyto nedostatky, které z časových důvodů nebyly zvládnutelné :

- Do zabezpečených částí aplikace, jako je například administrátorské rozhraní nebo rozhraní pro přihlášené uživatele, vyžadovat přístup pouze přes HTTPS protokol, který přenášená data ochrání proti zneužití.
- Jak uživatelské, tak administrátorské menu je třeba rozšířit o funkce dovolující jeho uživatelům editaci osobních údajů.
- Umožnění komunikace lékaře a pacienta přes jejich uživatelská rozhraní v systému.
- Optimalizovat obsah pro vyhledávače. Tato vlastnost určitě není stěžejní pro web typu nemocnice, ale s ohledem na vzrůstající počet uživatelů Internetu se i tyto stránky optimalizovat vyplatí.
- Přidat možnost upozorňování na mail či mobil při změně stavu objednávky, případně jen jako upozornění na blížící se termín návštěvy.
- Propojit správu aktualit s administrátorským rozhraním, aby bylo vše přístupné z jednoho prostředí.

Použitá literatura

- [1] Apache http server - *Wikipedie, otevřená encyklopedie*. [online] 3. březen 2008.
Dostupný z WWW: <http://cs.wikipedia.org/wiki/Apache_HTTP_Server>
- [2] JavaScript - *Wikipedie, otevřená encyklopedie*. [online] 22. dubna 2008.
Dostupný z WWW: <<http://cs.wikipedia.org/wiki/JavaScript>>
- [3] XSLT - *Wikipedie, otevřená encyklopedie*. [online] 22. dubna 2008.
Dostupný z WWW: <<http://cs.wikipedia.org/wiki/XSLT>>
- [4] MySQL - *Wikipedie, otevřená encyklopedie*. [online] 20. dubna 2008.
Dostupný z WWW: <<http://cs.wikipedia.org/wiki/Mysql>>
- [5] PHP - *Wikipedie, otevřená encyklopedie*. [online] 3. březen 2008.
Dostupný z WWW: <<http://cs.wikipedia.org/wiki/Php>>
- [6] Webový server - *Wikipedia, the free encyclopedia*. [online] 5. březen 2008.
Dostupný z WWW: <http://cs.wikipedia.org/wiki/Webov%C3%BD_server>
- [7] GRUDL, David. Konec W3C prostřednictvím psa II. *La Trine*. [online] 20. dubna 2008.
Dostupný z WWW: <<http://latrine.dgx.cz/konec-w3c-prostrednictvim-psa-ii>>
- [8] Nebojte se standardů! - *INTERVAL*. [online] 20. dubna 2008.
Dostupný z WWW: <<http://interval.cz/clanky/nebojte-se-standardu/>>
- [9] Přehled standardů W3C. *INTERVAL* [online] 20. dubna 2008.
Dostupný z WWW: <<http://interval.cz/clanky/prehled-standardu-w3c/>>
- [10] Luboslav Lacko: Oracle – Správa programování a použití databázového systému. *Computer Press*, 2003. ISBN: 80-7226—699-3
- [11] PROKOP M. Přehled standardů W3C. *INTERVAL* [online].
Dostupný z WWW: <<http://interval.cz/clanky/prehled-standardu-w3c/>>
- [12] HOULETTE F. *SQL-příručka programátora*. Praha : SoftPress, 2004.
- [13] KOSEK J. *PHP – tvorba interaktivních internetových aplikací: podrobný průvodce*. Praha: Grada, 1999.
- [14] JANOVSKEJ D. *Jak psát web*. [online] 13. březen 2008.
Dostupný z WWW: <<http://www.jakpsatweb.cz/>>

Příloha A Administrátorská dokumentace

Administrátorské rozhraní slouží k pohodlnému nastavení všech služeb, které systém poskytuje.

Pracoviště

Představuje libovolný pracovní útvar, u nějž máme tyto možnosti:

- **Název** – pojmenovává konkrétní pracovní útvar. Tento údaj bývá často vypisován v základním přehledu o pracovišti nebo jako položka menu.
- **Typ pracoviště** – označuje, o jaký druh pracoviště se jedná. Nutné pro univerzálnost a rozšiřitelnost systému. Mezi standardní typy pracoviště patří například Ambulance nebo Oddělení.
- **Nadřazené pracoviště** – umožní vybrat vytvářenému pracovišti jeho nadřazené pracoviště. Tímto krokem se zajišťuje hierarchie všech pracovišť.
- **Umístění** – je libovolný text s obrázkou, který popisuje, kde se pracoviště nachází.
- **Charakteristika** – je text s obrázkou charakterizující dané pracoviště.
- **Služby** – představují přehled výkonů a nabízených služeb na pracovišti.

K dispozici pro práci s pracovištěm jsou tyto funkce:

- **Vytvoření** – nového pracoviště.
- **Editace** – údajů již vytvořeného pracoviště.
- **Odstranění** – vybraného pracoviště.

Všechny tyto informace se pak můžou vypsát návštěvníkovi stránek.

Typ pracoviště

Nabízí tyto volby pro práci s typem pracoviště:

- **Vytvoření** – nového typu pracoviště.
- **Editace** – již vytvořeného typu.
- **Odstranění** – typu pracoviště.

Typ pracoviště konkretizuje obecnou položku pracoviště a zároveň umožňuje případnou rozšiřitelnost či pozdější úpravy. Pokud by v budoucnu došlo ke změně názvu nebo celkové restrukturalizaci, je vše připraveno k tomu, tyto události přijmout.

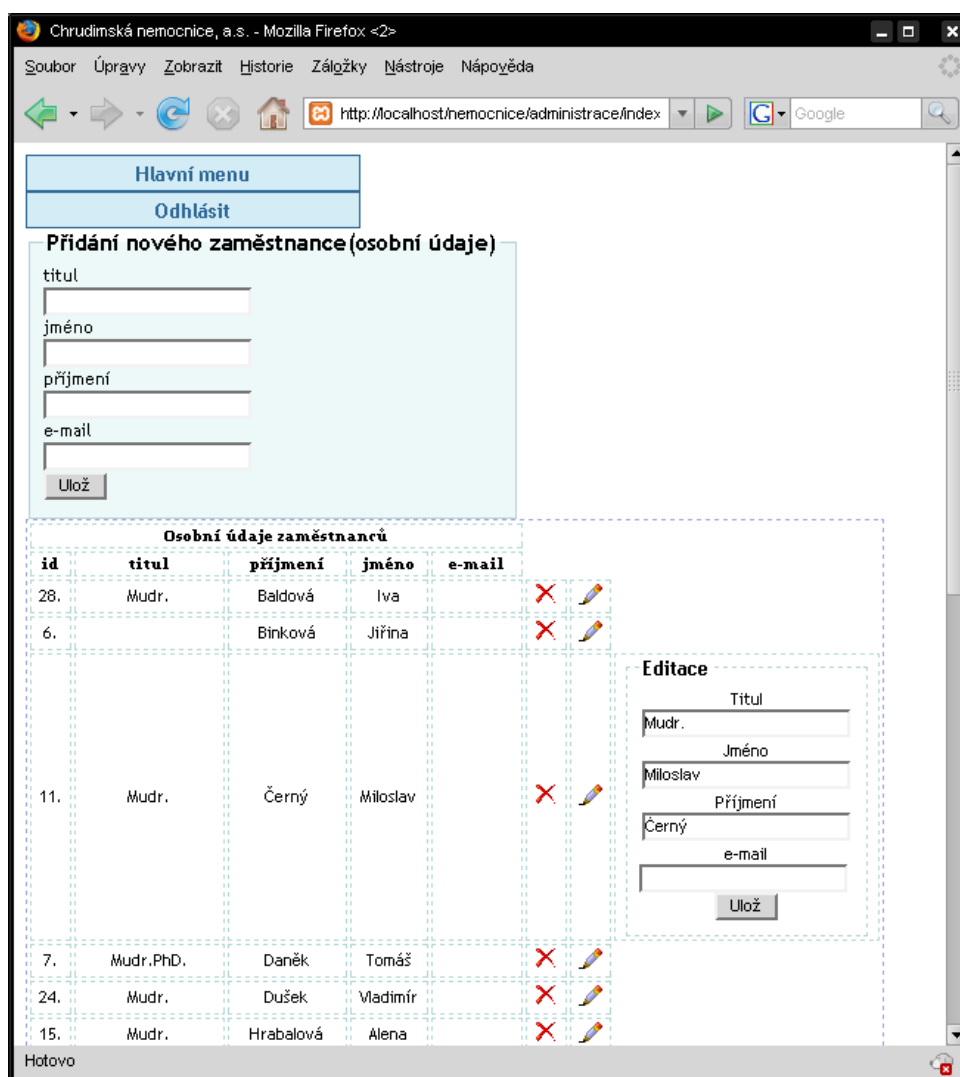
Osobní údaje zaměstnanců

Slouží k uchování těchto údajů:

- Jméno,
- příjmení,
- titul,
- email.

Pro práci s osobními údaji je k dispozici standardní paleta nástrojů:

- **Vytvoření** – nového zaměstnance.
- **Editace** – umožňuje libovolnou úpravu osobních údajů již vytvořených zaměstnanců.
- **Odstranění** – vybraného zaměstnance.



Osobní údaje – administrátor

Vytvoření pracovní funkce

Pracovní funkce slouží k pojmenování pracovních činností zaměstnanců.

K dispozici pro práci s pracovními funkcemi máme tyto volby:

- **Vytvoření** – po zadání názvu je možné vytvořit novou pracovní funkci.

- **Editace** – umožňuje libovolnou úpravu již vytvořených pracovních funkcí.
- **Odstranění** – zruší vybranou pracovní funkci.

Pracovní údaje

Pracovní údaje přiřazují osobní údaje zaměstnanců ke konkrétnímu pracovišti. Tato informace tedy říká, jaký zaměstnanec vykonává danou funkci na konkrétním pracovišti. Například zaměstnanec Jiří Novák pracuje na Interním oddělení ve funkci primáře oddělení.

Nastavení výše zmíněných parametrů umožní tyto volby:

- **Pracoviště** – Zde se určí, na jakém pracovišti se budou provádět změny.
- **Zaměstnanec** – Vybírá konkrétního zaměstnance.
- **Pracovní funkce** – Přiděluje ke konkrétnímu zaměstnanci na pracovišti jeho pracovní funkci.

V případě následných úprav jsou k dispozici standardní nástroje:

- **Editace** – umožňuje libovolnou úpravu pro již vytvořené pracovní údaje.
- **Odstranění** – odstraní vybraný pracovní údaj.

Správa uživatelských oprávnění

Umožňuje přidělit libovolnému zaměstnanci přihlašovací údaje a práva, která mu umožní podle úrovně oprávnění, přístup do určitých sekcí administrace. K vytvoření nového oprávnění je třeba nastavit tyto volby:

- **Uživatel** – Zajišťuje výběr konkrétního uživatele, kterému se mají přidělovat práva.
- **Login** – Slouží k jednoznačné identifikaci přihlašujícího se uživatele.
- **Heslo** – V databázi bezpečně uloženo hashovací funkcí SHA1. Při práci s hesly se vždy využívá výsledný hash a samotné heslo je tak chráněno.
- **Úroveň oprávnění** – Umožňuje rozdělit uživatele v základní verzi do tří skupin:
 - **Úroveň 1** – Oprávnění pro skupinu zdravotníků. Nejnižší úroveň, která uživateli poskytuje funkce pro práci s objednávkami, a to pouze na pracovišti, pod které spadá daný uživatel.
 - **Úroveň 2** – Oprávnění pro skupinu správce pracoviště. Tato úroveň umožňuje správu celého pracoviště, například editace informací o pracovišti nebo vytvoření ordinační doby. Všechny úkony lze provádět pouze v rámci pracoviště, pod které daný správce spadá.

▣ **Úroveň 3** – Oprávnění pro skupinu administrátor. Nejvyšší úroveň umožňující úpravy jakýchkoliv informací bez omezení.

Pomocí dostupných nástrojů je možné provádět:

- **Odstranění** – Odstraní vybraného uživatele.
- **Editace** – Umožní editaci přihlašovacích údajů o uživateli.

Kontakty

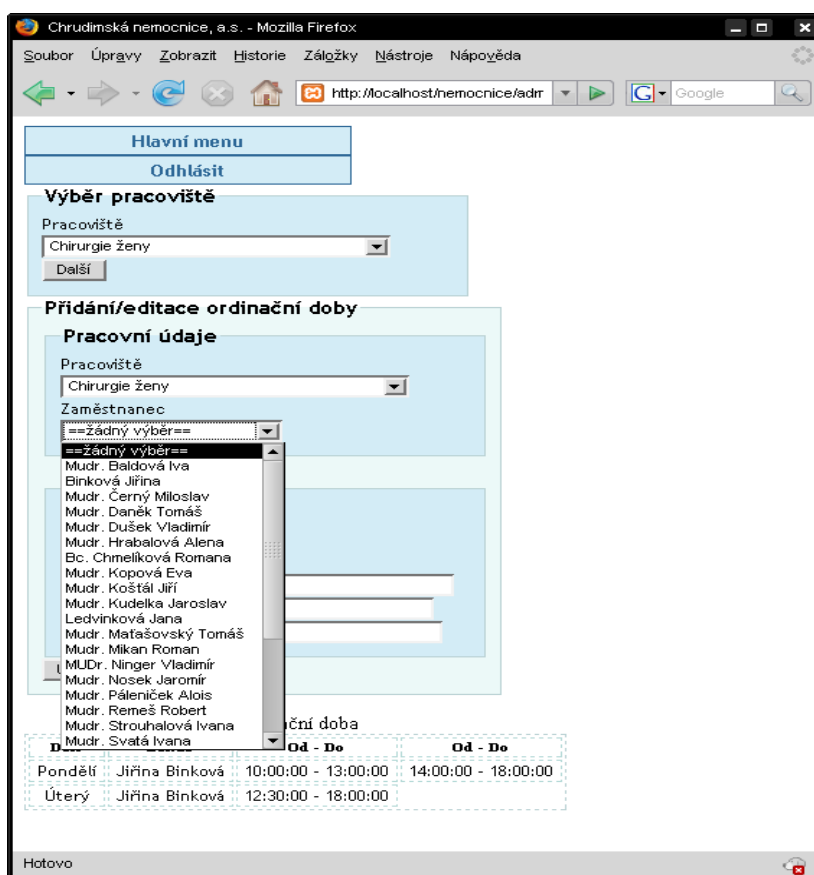
Jelikož zaměstnanci pracují na více pracovištích a na každém mají různá telefonní čísla, je nutné je uchovat tak, aby tato informace mohla být v budoucnu využita. K tomu slouží sekce kontakty, umožňující libovolnému pracovníkovi uchovat dvě telefonní čísla ke každému pracovišti, na kterém je zaměstnán.

Ordinační doba

K libovolnému pracovišti je možno také přiřadit ordinační dobu, kterou vytvoříme po zadání těchto údajů:

- **Pracoviště** – Určuje, k jakému pracovišti bude údaj o ordinačních hodinách přiřazen.
- **Zaměstnanec** – Výběr pracovníka, který má ordinovat.
- **Den** – Výběr dne v týdnu.
- **Od** – Nastavení, od kdy začíná ordinační doba.
- **Do** – Nastavení, do kdy trvá ordinační doba.
- **Čas pro pacienta** – Určuje dobu vyhrazenou pro jednoho pacienta.

Údaje o ordinační době jsou využívány pro vytváření objednávek, výpis volných termínů a k vytvoření tabulky s ordinačními hodinami pro dané pracoviště.



Ordinační doba – administrátor

Objednávky

Umožňují pověřenému zaměstnanci s administrátorskými právy zobrazit přehled uskutečněných objednávek pro vybrané pracoviště, případně objednávky rušit. Objednávky je možné:

- **Odstranit** – zruší vybranou objednávku.

Zrušení termínů pro objednání

V případě, že administrátor potřebuje zrušit na určité datum a na vybraném pracovišti termíny pro objednání, musí nastavit tyto volby:

- **Datum**, které určuje, v jakém termínu bude zrušení platné.
- **Ordinační doba** reprezentovaná příjmením zaměstnance ordinujícího v daný den, dnem v týdnu a časem, ve kterém začíná ordinační doba.

Typ položky menu

Pro automatické generování menu je důležité vědět, z čeho se bude generovat. A k tomu slouží právě typ položky menu, který říká, z jaké tabulky v databázi se bude vybírat objekt pro generování položek. Pokud by se v budoucnu rozšiřovala

databáze o nové tabulky, jejichž data by se měla zobrazovat návštěvníkům webu, jednoduše se přidá typ menu a můžeme s novou tabulkou ihned pracovat a generovat její obsah. Typ menu můžeme:

- **Vytvářet** – po zadání názvu tabulky, která reprezentuje daný typ.
- **Editovat** – změna názvu již vytvořeného typu menu.

Vlastnosti položky menu

Každá položka v menu reprezentuje určitou skupinu dat v databázi a vlastnosti položky menu zajistí výběr jen těch, které se mají zobrazit. Takže například u pracoviště Chirurgie budeme chtít zobrazit jen název a ordinační dobu a zbytek vlastností nás nezajímá.

Při zadání nové vlastnosti musíme zadat:

- **Název** – libovolné, ale výstižné pojmenování vlastnosti menu. Při práci s menu se bude vždy zobrazovat právě tento údaj.
- **Hodnota** – určuje buď hodnotu sloupce v tabulce, pokud existuje, nebo popisuje název speciální funkce.
- **Typ objektu** – nastaví, nad kterou tabulkou se bude hledat daná vlastnost.

K případným úpravám je k dispozici:

- **Editace** – umožní změnit hodnoty již vytvořených položek.

Menu

Představuje administraci základního ovládacího prvku, který návštěvníkovi bude zpřístupňovat žádané informace. Prvním krokem se vytvoří jednoznačná definice pozice a názvu položky v menu, k čemuž slouží nastavení těchto voleb:

- **Název** – pojmenovává položku menu. Tento údaj bude vždy zobrazen .
- **Výběr typu položky** – slouží k navolení, v jaké tabulce se bude hledat konečný objekt, který by měl reprezentovat danou položku. Tato možnost je tu pro případné rozšíření databáze o nové tabulky, které by byly použity ke generování menu a jeho obsahu. Základní nastavení nabízí tyto varianty :
 - Pracoviště – Konečný objekt bude vybírán z tabulky „pracoviste“.
 - Statický text – je tu pro případ, že je potřeba po kliknutí na položku v menu zobrazit libovolný text. Tento text je pak ukládán do tabulky „staticky_text“.
 - Typ pracoviště – reprezentuje tabulku „typ_pracoviste“, která obsahuje typy pracovišť. Tuto volbu je vhodné použít pro pozdější generování podmenu.

- **Výběr nadřazené položky** – umožní zvolit přímo nadřazenou položku, čímž se zajistí hierarchie celého menu.

Druhý krok poskytuje nastavení obsahu, který se zobrazí návštěvníkovi po kliknutí na vybranou položku. K dispozici jsou tyto možnosti nastavení:

- **Výběr konečného objektu** – nabídne existující položky z vybraného typu menu.
 - Nastavení obsahu – vygeneruje možné vlastnosti, které jsou dostupné pro daný typ pracoviště. Tyto vlastnosti lze samozřejmě přidávat a odebírat v závislosti na změnách v databázi.
- **Nastavení statického textu** – umožňuje zadat libovolný text s obrázky a tabulkami. Toto nastavení se zobrazí pouze pokud v předchozím kroku byl vybrán jako typ položky statický text.

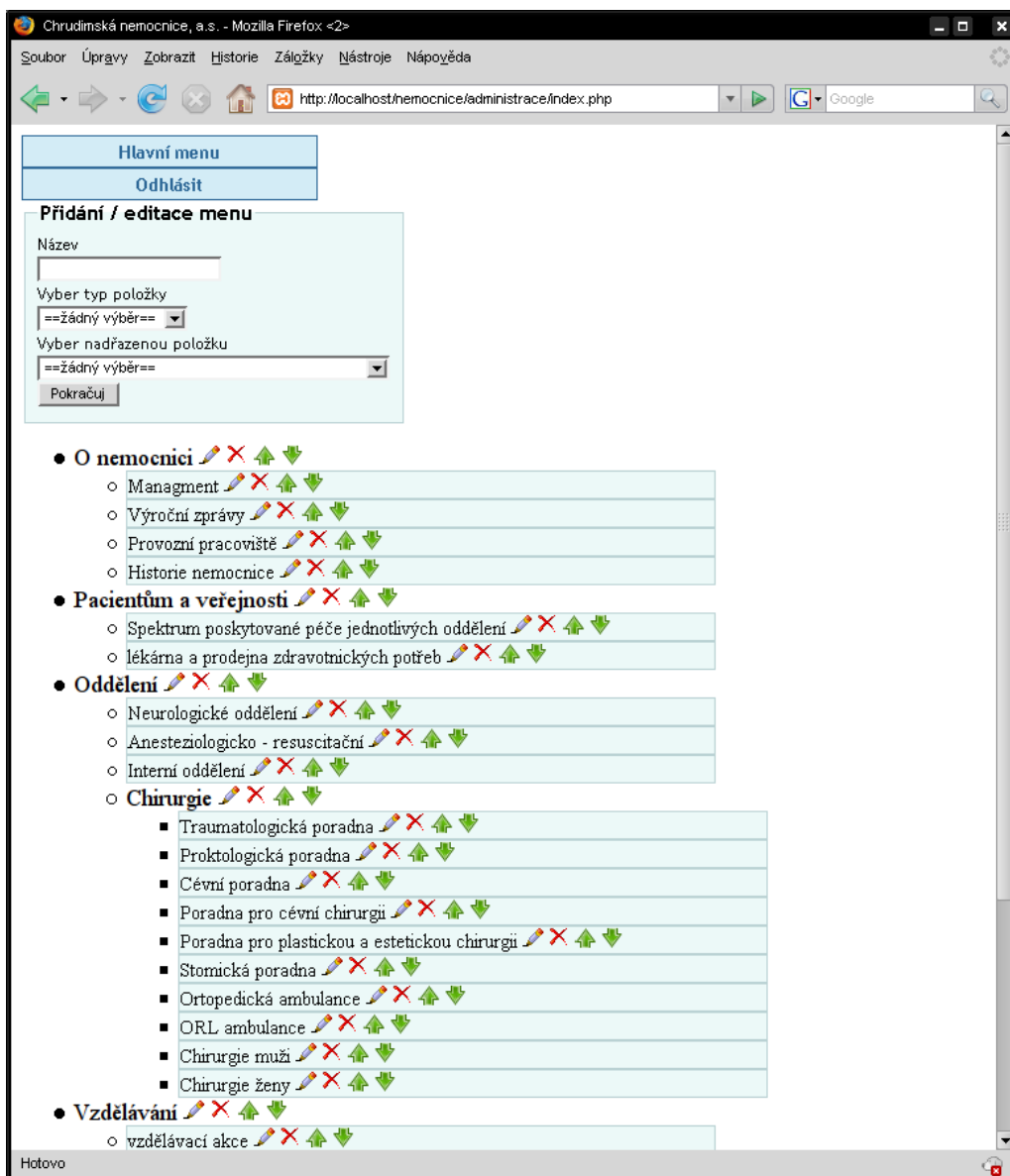
Třetí krok nabízí možnost generování podsložek podle typu položky. K úspěšnému vygenerování je třeba nastavit tyto volby:

- **Typ generovaného podmenu** – říká v jaké tabulce se bude hledat objekt, který je rodičem všech generovaných položek. Je možné generovat jednotlivé typy pracoviště například Ambulance, Oddělení. Nebo samotné pracoviště, například generování všech pracovišť, které spadají pod Chirurgii.
- **Výběr objektu** – umožní vybrání konkrétního objektu pro generování jeho podsložek.
- **Nastavení obsahu podsložek** – nastavuje všem generovaným objektům vlastnosti, které se mají zobrazovat.

Nástroje pro práci s menu nám umožní tyto akce:

- Editace vybrané položky menu,
- úprava položky menu,
- posun položky menu o jednu nahoru,
- posun položky menu o jednu dolů,
- odstranění položek menu,

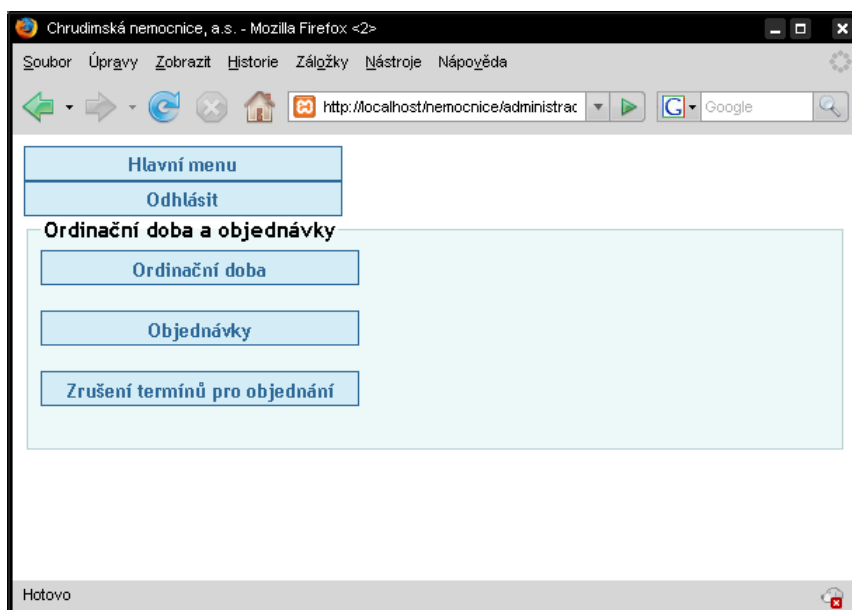
- přidání nové položky menu.



Menu – administrátor

Příloha B Dokumentace správce pracoviště

Rozhraní správce pracoviště slouží k modifikaci určitého pracoviště a informací s ním spojených. Správce pracoviště může užívat stejné nástroje pro úpravu pracoviště jako administrátor s tím rozdílem, že nemá možnost volby pracoviště a veškeré možnosti jsou omezeny pouze na přidělené pracoviště, které je nastaveno v pracovních údajích.



Volby – rozhraní správce pracoviště

Ordinační doba

Správce pracoviště může vytvářet a modifikovat ordinační dobu pouze na pracovišti, na které patří. Veškeré volby jsou též omezeny na konkrétní pracoviště a s tímto omezením také vypisovány.

K úspěšnému vytvoření ordinační doby je třeba zadat tyto údaje:

- **Zaměstnanec** – představuje volbu pracovníka, který má ordinovat. Výběr zaměstnance je omezen pouze na ty, kteří pracují na konkrétním pracovišti.
- **Den** – reprezentuje den v týdnu.
- **Od** – umožňuje nastavení času, od kdy začíná ordinační doba.
- **Do** – umožňuje nastavení času, do kdy trvá ordinační doba.
- **Čas pro pacienta** – určuje dobu vyhrazenou pro jednoho pacienta.

Údaje o ordinační době jsou využívány pro vytváření objednávek, výpis volných termínů a k vytvoření tabulky s ordinačními hodinami pro dané pracoviště.

Hlavní menu
Odhlásit

Přidání/editace ordinační doby

Pracovní údaje

Zaměstnanec
 ==žádný výběr==
 Binková Jiřina
 Mudr. Černý Miloslav
 Mudr. Vítavský Tomáš

Den
 ==žádný výběr==

Od (formát: př. 10:00:00)

Do (formát: 12:30:00)

Čas pro pacienta(min)

Ulož

Ordinační doba

Den	Lékař	Od - Do	Od - Do
Pondělí	Jiřina Binková	10:00:00 - 13:00:00	14:00:00 - 18:00:00
Úterý	Jiřina Binková	12:30:00 - 18:00:00	

Hotovo

Ordinační doba – správce pracoviště

Objednávky

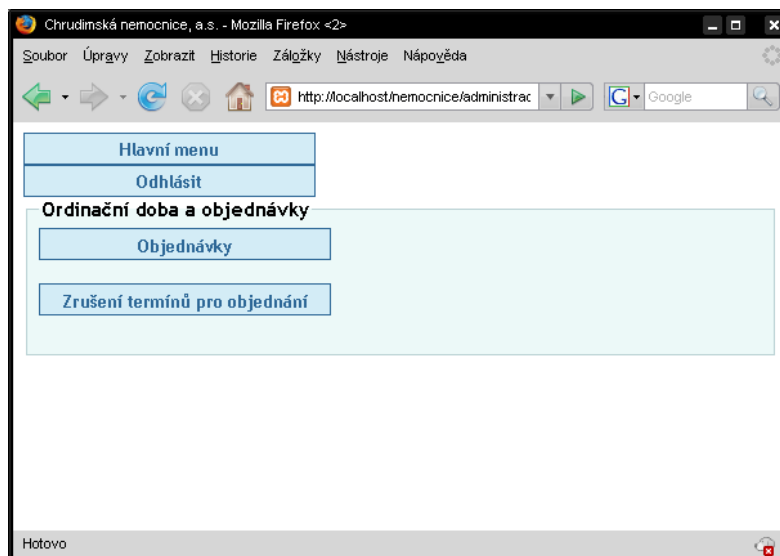
Stejně možnosti jako u administrátorského rozhraní s tím rozdílem, že pracoviště není možno libovolně volit, nýbrž je automaticky přiděleno dle nastavení pracovních údajů a podle toho jsou také vypsány objednávky.

Zrušení termínů pro objednání

Umožňuje uživateli zrušit termíny pro objednání stejně jako v administrátorském režimu s tím rozdílem, že nelze vybrat libovolnou ordinační dobu, ale jsou k dispozici pouze ty, které se vztahují k pracovišti z pracovních údajů správce pracoviště.

Příloha C Dokumentace zdravotníka

Úroveň zdravotníka disponuje nejnižšími právy, umožňující pouze základní úkony na jeho pracovišti.



Volby – rozhraní zdravotního pracovníka

Objednávky

Nabízí stejné možnosti jako u rozhraní správce pracoviště. Veškeré volby jsou vypisovány s omezením na konkrétní zdravotníckovo pracoviště.

Zrušení termínů pro objednání

Nabízí stejné možnosti jako u rozhraní správce pracoviště. Veškeré volby jsou vypisovány s omezením na konkrétní zdravotníckovo pracoviště.

Příloha D Uživatelská dokumentace

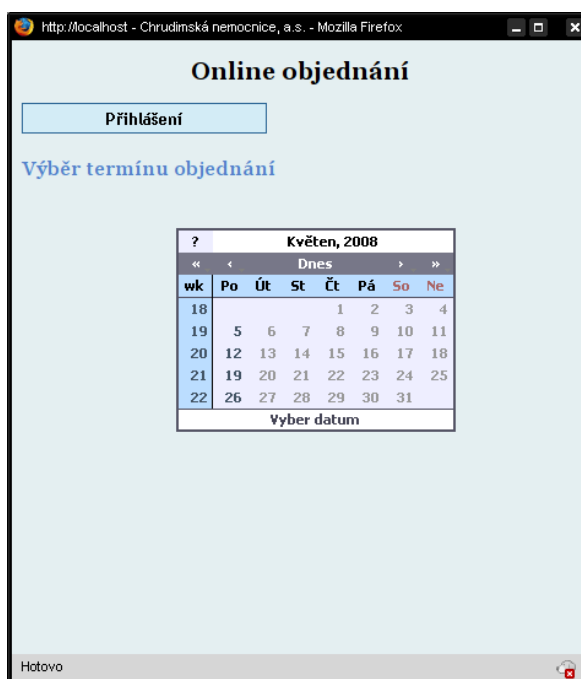
Uživatel neboli návštěvník stránek se může pohybovat po webu a zobrazovat libovolné informace. V případě, že projeví zájem o objednání k lékaři, je přenesen do objednávkového systému s těmito volbami:

Online objednání

V případě, že pracoviště nabízí online objednání, má návštěvník webu možnost uskutečnit objednávku.

Výběr data objednání

Poskytuje ve formě jednoduchého kalendáře přehled termínů ordinačních dob, na které je možné se objednat. V případě, že se v daný den neordinuje, je políčko znepřístupněno. Aktivní dny jsou kvůli lepší orientaci zvýrazněny. Po vybrání konkrétního data se vyvolá seznam volných časových intervalů a pracovníků, kteří ordinují ve zvolené datum na vybraném pracovišti.



Kalendář - pacient

Výběr časového intervalu a lékaře

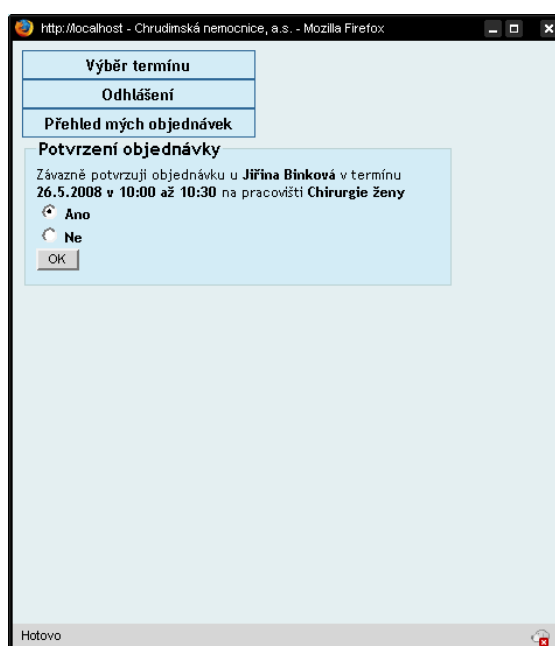
Umožňuje výběr volného časového úseku a pracovníka, který v tu dobu ordinuje. Po výběru časového intervalu a lékaře je zobrazen přehled celé objednávky s výzvou na dokončení.



Výběr termínu – pacient

Dokončení objednávky

Pokud je pacient úspěšně přihlášen do systému, zobrazí se mu informace o všech detailech jeho objednávky s možností dokončit objednávku nebo ji zrušit. Potvrzením provede dokončení objednávky.



Dokončení objednávky – pacient

Přehled mých objednávek

Po přihlášení pacienta do systému se objeví nová položka – přehled mých objednávek, která zobrazí všechny pacientovy aktuální objednávky.

Přihlášení

Umožní zaregistrovanému uživateli po zadání správného přihlašovacího jména a hesla přístup do systému. V případě, že je to nový uživatel, je mu nabídnuta registrace.

Registrace

Slouží k vytvoření uživatelského účtu, pomocí nějž pacient vytváří nové objednávky a sleduje přehled již vytvořených.

K úspěšné registraci je nutné zadat tyto údaje:

- Titul,
- jméno,
- příjmení,
- email,
- telefon,
- přihlašovací jméno,
- heslo.

Příloha E Programátorská dokumentace

Pro funkčnost celé aplikace je nutné mít server s php modulem a MySQL databází. Celá aplikace je uložena na přiloženém CD. V kořenovém adresáři se nachází složka *install*, která obsahuje soubor *databaze.dat*, což je vyexportovaná MySQL databáze a adresář *nemocnice*. Pro přístup na server a do importované databáze jsou použity tyto údaje:

- Uživatel: root,
- heslo: rootroot,
- hostitel: localhost,
- název databáze: nemocnice.

V případě, že bude nutné provést změnu těchto údajů, je třeba editovat v souborech *fce_pro_vsechny.php* a *fce_admin.php*, obsažené v adresáři *nemocnice*, funkce *spojeni()* a *spojeni_admin()*.

Dále je nutné nakopírovat adresář *nemocnice* na webový server a nastavit mu vhodná práva. Poté je vše připraveno ke spuštění aplikace, která by pro lokální běh měla být přístupná z adresy <http://localhost/nemocnice/index.php> jako hlavní prezentace a administrátorská část z <http://localhost/nemocnice/administrace/index.php>.

Pro testovní účely jsou pro zaměstnanecké rozhraní vytvořeny tyto uživatelská přihlašovací jména:

- administrator,
- spravce,
- zdravotnik.

Jejich hesla jsou nastavena na stejné hodnoty, jako jsou přihlašovací jména.

Pro testování pacienta je možné vytvořit vlastní registraci a otestovat objednávání u Chirurgického oddělení muži nebo ženy, kde jsou vypsány volné termíny.