

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství



Bakalářská práce

Využití QR kódu pro orientaci v budově

Karolína Pikorová

© 2018 ČZU v Praze

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Karolína Pikorová

Informatika

Název práce

Využití QR kódu pro orientaci v budově

Název anglicky

QR code use for orientation in the building

Cíle práce

Cílem práce je navrhnout, vytvořit a vyzkoušet informační systém založený na štítcích s QR kódy a mobilní aplikací, který bude sloužit k orientaci v budově PEF. Uživatel bude moci zadat místo, kam se potřebuje dostat a po přečtení nejbližšího QR kódu dostane informaci o cestě na dané místo. Bude možné zadávat osoby v kancelářích, názvy pracovišť a učebny.

Metodika

Nejprve bude sestaven popis funkčních požadavků na ukázkovou aplikaci podle standardu UML a dalších norem softwarového inženýrství. Součástí řešení bude výběr vhodné mobilní platformy (iPhone, Android, Windows). Výsledná implementace bude dokumentována a otestována na malém vzorku reálných dat katedry informačního inženýrství a vybraných vchodů a schodišť v budově.

Doporučený rozsah práce

30-40 stran

Klíčová slova

QR kod, mobilní aplikace, platforma iPhone, platforma Android, platforma Windows, orientace v budovách

Doporučené zdroje informací

DUCKETT, Jon. HTML & CSS: design and build websites. Indianapolis, IN: Wiley, c2011. ISBN 1118008189.

HERMES, Dan. Xamarin Mobile Application Development: Cross-Platform C# and Xamarin. Forms Fundamentals. California: Apress, 2015. ISBN 9781484202159.

SHACKLES, Greg. Mobile development with C#. Sebastopol, CA: O'Reilly, c2012. ISBN 1449320236.

WATERS, Joe. QR codes for dummies. Portable ed. Hoboken, NJ: John Wiley, c2012. –For dummies. ISBN 1118337034.

Předběžný termín obhajoby

2017/18 LS – PEF

Vedoucí práce

doc. Ing. Vojtěch Merunka, Ph.D.

Garantující pracoviště

Katedra informačního inženýrství

Elektronicky schváleno dne 11. 1. 2018

Ing. Martin Pelikán, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 11. 1. 2018

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 23. 02. 2018

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Využití QR kódu pro orientaci v budově" jsem vypracovala samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autorka uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 15. 3. 2018

Poděkování

Ráda bych touto cestou poděkovala doc. Ing. Vojtěchu Merunkovi, Ph.D. za odborné vedení a odbornou pomoc při zpracovávání této práce a blízkým přátelům a rodině za podporu, která mi při psaní práce pomáhala.

Využití QR kódu pro orientaci v budově

Abstrakt

Tato bakalářská práce se zabývá problematikou mobilních aplikací spolupracujících pomocí čtečky s QR kódy. Analyzuje a popisuje zvolené mobilní platformy – iOS, Android a Windows Phone.

Hlavním cílem práce je navrhnout, vytvořit a vyzkoušet informační systém založený na štítcích s QR kódy. Návrh aplikace je vytvořen podle standardu UML a dalších norem softwarového inženýrství. Pro implementaci byl zvolen programovací skriptovací jazyk PHP, webový server Apache a systém řízení báze dat MySQL.

Výsledná vytvořená aplikace poskytuje uživatelům snadnou orientaci ve zvolené budově.

Klíčová slova: QR kód, mobilní aplikace, UML, PHP, platforma iPhone, platforma Android, platforma Windows, orientace v budovách

QR code use for orientation in the building

Abstract

This bachelor thesis deals with problems of mobile applications cooperating with a QR code using QR code reader. It analyses and describes selected mobile platforms - iOS, Android and Windows Phone.

The main aim of the thesis is to design, create and test the information system based on QR codes. The application design is based on UML standards and other standards of software engineering. The PHP programming scripting language, the Apache web server and the MySQL database management system were selected for implementation.

The resulting created application provides users with easy orientation in the chosen building.

Keywords: QR code, mobile application, UML, PHP, iPhone platform, Android platform, Windows platform, building orientation

Obsah

1 Úvod.....	11
2 Cíl práce a metodika	12
2.1 Cíl práce	12
2.2 Metodika	12
3 Teoretická východiska	13
3.1 World Wide Web	13
3.1.1 Rozhraní CGI.....	13
3.1.1.1 CGI-skripty.....	13
3.2 Webová aplikace	14
3.2.1 Klient / server architektura	14
3.2.1.1 Hyper Text Transfer Protokol	14
3.2.2 Klient	15
3.2.2.1 Webový prohlížeč.....	15
3.2.3 Server.....	15
3.2.3.1 Webový server Apache.....	16
3.2.3.2 IIS	16
3.3 Technologie na straně klienta/serveru.....	16
3.3.1 HTML	16
3.3.2 CSS	17
3.3.3 DHTML	17
3.3.4 JavaScript.....	17
3.3.5 ASP	18
3.3.6 PHP	18
3.3.6.1 Historie	18
3.4 Databáze	19
3.4.1 MySQL	19
3.4.2 PostgreSQL.....	20
3.5 UML – grafický modelovací jazyk	21
3.5.1 Historie.....	21
3.5.1.1 Konsorcium OMG	21
3.5.2 UML diagramy	21
3.5.2.1 Třídní diagram	22
3.5.2.2 Diagram případů užití.....	22
3.6 QR kódy	23

3.6.1	Přístup k informacím	23
3.6.1.1	Čtečka QR kódů	23
3.6.2	Srovnání QR kódů s tradičními čárovými kódy	23
3.6.3	Další 2D formáty čárových kódů.....	24
3.7	Mobilní platformy	26
3.7.1	iOS	26
3.7.2	Android	26
3.7.3	Windows Phone	27
4	Vlastní práce	28
4.1	Analýza	28
4.1.1	Plánované činnosti	28
4.1.2	Požadavky na aplikaci	28
4.1.3	Výběr vhodné mobilní platformy	28
4.2	Modelový návrh systému	29
4.2.1	Databázový model	29
4.2.2	Diagram případů užití	30
4.2.3	Objektový model.....	30
4.3	Návrh vizuální reprezentace.....	30
4.3.1	Návrh UI	30
4.3.1.1	Důležité fáze procesu tvorby UI.....	30
4.3.2	Návrh QR kódu	31
4.3.3	Návrh loga.....	31
4.4	Použité technologie	32
4.4.1	Programovací jazyk	32
4.4.2	Operační systém.....	32
4.4.3	System řízení báze dat	32
4.5	Implementace	33
4.5.1	Vývojové prostředí	33
4.5.2	Verzovací systém.....	33
4.5.3	Výsledné nasazení.....	33
4.6	Testování	34
4.6.1	Testovací scénáře	34
4.6.2	Testování zobrazení	34
4.6.3	Mobilní platformy.....	35
5	Výsledky a diskuse	36
6	Závěr.....	37
7	Seznam použitých zdrojů	38

8 Přílohy	40
------------------------	-----------

Seznam obrázků

Obrázek 1 - QR kód umožňuje ukládat více informací než čárový kód.....	23
Obrázek 2 - DataMatrix	24
Obrázek 3 - Aztécký kód	25
Obrázek 4 - PDF 417	25
Obrázek 5 - Datový model.....	29
Obrázek 6 - Vytvořené logo.....	31

1 Úvod

Tato bakalářská práce se zabývá problematikou mobilních aplikací spolupracujících pomocí čtečky s QR kódy a webových aplikací, které jsou pomocí QR kódu spustitelné.

V první části práce jsou popsána teoretická východiska potřebná pro vytváření webových aplikací. Jsou v ní popsány webové služby, komunikace po internetu a technologie používané při tvorbě webových aplikací. Závěr se zaměřuje především na popis a použití QR kódů a důležitou část, což je výběrový popis mobilních platforem.

Druhá část obsahuje analýzu nejpoužívanějších mobilních platforem na trhu – iOS, Android a Windows Phone. Následně analýzu a návrh nové aplikace spolupracující s QR kódy a výběr správného generátoru QR kódů. Výsledně vytvořená aplikace je veřejně nasazena jako prototyp.

2 Cíl práce a metodika

2.1 Cíl práce

Cílem práce je navrhnout, vytvořit a vyzkoušet informační systém založený na štítcích s QR kódy a mobilní aplikací, který bude sloužit k orientaci v budově PEF. Uživatel bude moci zadat místo, kam se potřebuje dostat a po přečtení nejbližšího QR kódu dostane informaci o cestě na dané místo. Bude možné zadávat osoby v kancelářích, názvy pracovišť a učebny.

2.2 Metodika

Nejprve bude sestaven popis funkčních požadavků na ukázkovou aplikaci podle standardu UML a dalších norem softwarového inženýrství. Součástí řešení bude výběr vhodné mobilní platformy (iPhone, Android, Windows). Výsledná implementace bude dokumentována a otestována na malém vzorku reálných dat katedry informačního inženýrství a vybraných vchodů a schodišť v budově.

3 Teoretická východiska

Kapitola popisuje základní stanoviska zabývající se technologiemi pro vývoj webových aplikací, prací s QR kódy a nejpoužívanějšími mobilními platformami a jejich rozborem.

3.1 World Wide Web

Služba World Wide Web (zkratka WWW) byla poprvé spuštěna v roce 1990 na půdě evropského výzkumného centra CERN. V tu dobu existovali tři technologie, se kterými se vystačilo. [1]

První z nich byl jazyk HTML (HyperText Markup Language), který sloužil k zápisu webových stránek. Druhou nezbytnou technologií byl přenosový protokol HTTP (HyperText Transfer Protocol), který zajišťuje přenos HTML stránek z WWW serveru do prohlížeče. Třetí technologií nezbytnou pro implementování služby WWW jsou URL (Uniform Resource Locator). Každý objekt přístupný na Webu má svoji jedinečnou URL adresu, která slouží k vytváření odkazů na daný objekt. [1]

3.1.1 Rozhraní CGI

Velkou inovací byla možnost automatického generování stránek, které obsahují informace proměnlivé v čase. HTML stránka je soubor uložený na disku WWW serveru, který má své URL. Nic však nebrání tomu, aby URL ukazovalo na nějaký spustitelný soubor neboli program, který HTML stránku vygeneruje. Tato stránka pak může obsahovat aktuální informace. Spustitelný soubor je vyvoláván WWW serverem a bylo proto zapotřebí rozhraní, které by definovalo způsob spuštění programu a předávání dat mezi WWW serverem a programem. [1]

Rozhraní se jmenuje CGI (Common Gateway Interface) a popisuje vše, co musí program splnit, aby z něj rozhraní bylo schopno vytvořit platnou webovou stránku. [2]

3.1.1.1 CGI-skripty

Programům, které generují HTML stránky, se proto často říká CGI-skripty. [1]

Pro psaní CGI-skriptů můžeme použít téměř libovolný programovací jazyk. Nejčastěji se používají různé příkazové interprety Unixu a jazyk Perl. Poměrně často se CGI-skripty píšou i v jazycích C nebo C++. [3]

Aby byl výsledný program použitelný jako CGI-skript, musí splňovat pouze dvě podmínky:

- Musí umět přebírat parametry předané pomocí rozhraní CGI.
- Výsledkem jeho činnosti je odpověď ve formátu HTTP zapsaná na standardní výstup programu. [3]

3.2 Webová aplikace

Aplikace, kterou není nutno instalovat na zařízení uživatele a může se spustit z kteréhokoliv zařízení pomocí webového prohlížeče, protože je spuštěna na straně serveru. Vzhledem k tomu, že je třeba jen prohlížeč se webová aplikace někdy nazývá též jako lehký klient. [4]

3.2.1 Klient / server architektura

Prakticky všechny služby internetu jsou poskytovány vždy dvojicí programů: programem typu server, který obstarává samotnou službu a programem typu klient, který zprostředkovává styk uživatele se serverem. Komunikace mezi klientem a serverem se děje podle přesných formálních pravidel, kterým se říká protokol. [5]

Protokol je soubor určitých pravidel, které musí dané zařízení nebo program splňovat, aby byl schopen komunikovat s jiným odpovídajícím zařízením nebo programem. Proces tvorby síťových protokolů je dlouhodobý a protokoly se neustále vyvíjí. Návrhy i definitivní standardy jsou zveřejňovány formou RFC dokumentů. Zkratka RFC pochází ze slov Request For Comment a RFC dokumenty jsou veřejně přístupné na internetu. [6]

3.2.1.1 Hyper Text Transfer Protokol

Hyper Text Transfer Protocolu (zkratka HTTP) je jeden z nejdůležitějších protokolů internetu pro přenos dat mezi serverem a klientem. Klienty HTTP představují většinou webové prohlížeče a HTTP servery jsou většinou webové servery. Před vznikem HTTP byl nejdůležitějším protokolem pro přenos souborů protokol FTP (File Transfer Protocol). [7]

HTTP používá jako některé další aplikace jednotný lokátor prostředků URL (Uniform Resource Locator), který specifikuje jednoznačné umístění nějakého zdroje v internetu. Veškerá komunikace u protokolu HTTP probíhá tak, že klient vyšle požadavek

a server na něj odpovídá. Základními operacemi protokolu HTTP je načtení webové stránky, získání informací o webové stránce a odeslání dat z formuláře či skriptu vloženého do WWW stránky na HTTP server. Tyto operace se provádí pomocí příkazů GET, HEAD a POST. [7]

Protokol HTTP komunikuje na portu 80 TCP protokolu, při zadávání adresy ve formátu URL je ovšem možno zadat i jiný port. Navíc protokol HTTP není závislý na protokolu TCP a může pracovat i s jinými spolehlivými protokoly. [7]

K protokolu HTTP existuje také jeho bezpečnější verze označovaná jako HTTPS, která umožňuje přenášet data šifrovat, a tím chránit před odposlechem či jiným narušením. [8]

HTTP je spolu s elektronickou poštou tím nejvíce používaným protokolem a zasloužil se o obrovský rozmach internetu v posledních letech. [8]

3.2.2 Klient

Klientem může být buď nějaká aplikace, nebo systém, který využívá služeb serveru. [5]

3.2.2.1 Webový prohlížeč

Webový prohlížeč je počítačový program, který slouží k prohlížení World Wide Webu. Program umožňuje komunikaci s HTTP serverem a zpracování přijatého kódu (HTML, XHTML, XML), který podle daných standardů zformátuje a zobrazí webovou stránku. Textové prohlížeče zobrazují obvykle stránky jako velmi jednoduše formátovaný text. Grafické prohlížeče umožňují složitější formátování stránky včetně zobrazení obrázků. Pro zobrazení některých zvláštních součástí stránky, jako jsou flashové animace nebo javové applety, je třeba prohlížeč doplnit o specializované zásuvné moduly. [9]

Zásuvné moduly (pluginy) jsou soubory, které zprostředkovávají spolupráci prohlížeče s externími programy. Díky nim lze přehrát video na webových stránkách, číst soubory různých formátů přímo ve webovém prohlížeči či stahovat data pomocí externích download managerů. [9]

3.2.3 Server

Když se webový prohlížeč dotazuje na webovou stránku, žádost se pošle přes internet do speciálního počítače známého jako webový server, který danou webovou

stránku hostuje. Webové servery jsou speciální počítače, které jsou neustále připojeny k Internetu a jsou optimalizovány pro odesílání webových stránek prohlížečům, které o ně požádají. [10]

Dva nejběžněji používané webové servery, které ovládají přibližně sedmdesát procent trhu, jsou webový server Apache a IIS (Internet Information Services). [11]

3.2.3.1 Webový server Apache

Webový server Apache je open source aplikace spravována skupinou Apache Software Foundation. Přestože je kompatibilní se všemi hlavními operačními systémy, je nejčastěji používána v kombinaci s Linuxem. [11]

3.2.3.2 IIS

IIS (Internet Information Services) je webový server společnosti Microsoft. Je součástí jejich operačního systému Windows a je kompatibilní pouze s ním. Jedná se o uzavřený softwarový produkt a je podporován výhradně společností Microsoft. [11]

Odlehčená verze IIS je IIS Express a je určena zejména pro vývojáře webových aplikací. [11]

3.3 Technologie na straně klienta/serveru

Všechny následující zmíněné technologie můžeme rozdělit do dvou skupin podle toho, zda jsou prováděny na serveru nebo na klientovi. Nic však nebrání jejich vzájemné kombinaci, pokud je účelná. [1]

3.3.1 HTML

HyperText Markup Language (zkratka HTML) je název značkovacího jazyka používaného pro tvorbu webových stránek, které jsou propojeny hypertextovými odkazy. HTML je hlavním z jazyků pro vytváření stránek v systému World Wide Web, která umožňuje publikaci dokumentů na internetu. [8]

Nejnovější verze HTML je HTML5, která ukončuje závislost HTML na SGML, opravuje mnoho chyb předešlé verze, vyřazuje mnoho zastaralých a již nepoužívaných prvků a přidává nové sémantické prvky. [12]

3.3.2 CSS

Cascading Style Sheets (zkratka CSS) je jazyk pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML. Hlavním smyslem je umožnit návrhářům oddělit vzhled dokumentu od jeho struktury a obsahu. [12]

Dnes se také běžně používají CSS preprocesory. CSS preprocesor je jazyk, který je postavený nad CSS. Přidává do něj nové jazykové vlastnosti nebo řeší jiné jeho technické slabiny. CSS preprocesor je nástroj, který ze zdrojového kódu zapsaného ve vlastní syntaxi vygeneruje CSS pro prohlížeč. Mezi nejznámější patří SASS, LESS a Stylus. [13]

3.3.3 DHTML

Microsoft v roce 1997 přišel ve svém prohlížeči MSIE 4.0 s novou technologií dynamického HTML (DHTML). DHTML vychází ze stejné myšlenky jako JavaScript, ale mnohem více ji dotahuje. V DHTML je přístupný každý objekt webové stránky a s každým objektem můžeme kdykoliv manipulovat — měnit jeho obsah, způsob zobrazení či polohu. Jako objekty jsou přístupné i styly připojené ke stránce a jejich změnou lze velice jednoduše dosáhnout překvapivých vizuálních efektů. Teprve stránky zapsané pomocí DHTML jsou skutečně interaktivní. Velkou výhodou DHTML je zachování obousměrné kompatibility s běžnými HTML stránkami. [1]

3.3.4 JavaScript

Novou technologií, která vznikla v roce 1996, byl JavaScript. JavaScript je jednoduchý objektově orientovaný skriptovací jazyk se syntaxí vycházející z jazyku Java. S JavaScriptem přišla firma Netscape a zabudovala jej do svého legendárního prohlížeče Netscape Navigator. [1]

JavaScript se zapisoval přímo do HTML kódu stránky a uměl posloužit v mnoha situacích. Jeho nejčastější použití bylo ve spojení s formuláři. Krátké skripty v JavaScriptu mohly kontrolovat správnost údajů v polích formuláře ještě před odesláním na server. Uživatel tak získal nesrovnatelně rychlejší odezvu v porovnání s klasickým způsobem využívajícím pouze CGI-skripty. Druhou oblastí použití JavaScriptu byla drobná vylepšení interaktivnosti stránek. [1]

3.3.5 ASP

ASP (Active Server Pages) je skriptovací platforma společnosti Microsoft, primárně určená pro dynamické zpracování webových stránek na straně serveru. [8]

ASP technologie je v současné době zastaralá a nahrazená technologií ASP.NET. ASP.NET je součástí .NET Frameworku pro tvorbu webových aplikací a služeb. ASP.NET, lze chápat jako širší a komplexnější technologii, která se od ASP v mnoha ohledech fundamentálně liší. [8]

3.3.6 PHP

PHP (původně Personal Home Page, nyní obvykle rekurzivně Hypertext Preprocessor) je široce používaný open source skriptovací programovací jazyk, určený především pro programování dynamických internetových stránek. [14]

Zatím poslední vydanou verzí PHP je verze 7. [15]

Nejčastěji se začleňuje přímo do struktury jazyka HTML, XHTML či XML, což lze využít při tvorbě webových aplikací. PHP lze použít i k tvorbě konzolových a desktopových aplikací. PHP je vedle ASP jedním ze dvou nejrozšířenějších skriptovacích jazyků pro web. Oblíbeným se stal především díky jednoduchosti použití, bohaté zásobě funkcí, a tomu, že kombinuje vlastnosti více programovacích jazyků, a nechává tak vývojáři částečnou svobodu v syntaxi. [8]

Další výhodou PHP je jeho nezávislost na platformě. Dnes jsou k dispozici verze PHP pro Unix a Windows. PHP není svázáno s žádným konkrétním serverem, může běžet na libovolném. Nejlépe si však dnes rozumí se serverem Apache. [1]

3.3.6.1 Historie

Na počátku zrodu systému stál Rasmus Lerdorf. Psal se rok 1994 a Rasmus si ve volném čase vytvořil v Perlu jednoduchý systém pro evidování přístupu k jeho stránkám. Jelikož neustálé spouštění interpretu Perlu velmi zatěžovalo WWW server, přepsal autor systém do jazyka C. [1]

Ačkoliv byl celý systém původně určen pro osobní Rasmusovo použití, zalíbil se i ostatním uživatelům serveru a začali ho používat. Po čase přicházeli s požadavky na vylepšení celého systému. Autor proto systém rozšířil, doplnil o dokumentaci a uvolnil jej

pod názvem Personal Home Page Tools, který se později změnil na Personal Home Page Construction Kit. [1]

Kromě již zmíněného systému pro evidování přístupů ke stránkám vytvořil pan Lerdorf i nástroj, který umožňoval začleňování SQL-dotazů do stránek, vytváření formulářů a zobrazování výsledků dotazů. Program, který umožnil zpřístupnění databázi na Webu, se jmenoval Form Interpreter (FI). [1]

Vývoj PHP však není zdaleka ukončen. Do jazyka jsou přidávány stále nové možnosti, které umožňují efektivnější vytváření internetových aplikací. [1]

3.4 Databáze

Databázi si můžeme představit jako místo, kam se ukládají všechny potřebné údaje. Přístup k údajům uloženým v databázi obstarává program, kterému se říká SŘBD — Systém Řízení Báze Dat. Mezi SŘBD patří takové programy jako Oracle, MS SQL Server, Sybase, Informix, Progress či InterBase. Naštěstí i na poli SŘBD existují programy šířené zdarma jako freeware — mSQL, MySQL a PostgreSQL. Převážná většina dnes používaných SŘBD při uspořádání údajů v databázi vychází z relačního modelu dat. [1]

Název tohoto modelu vychází z relační algebry, což je matematický aparát, na kterém relační model dat staví. V tomto modelu jsou údaje uspořádány do tabulek. Tabulka zpravidla shromažďuje údaje o jednom druhu objektů. [1]

Způsob komunikace se SŘBD je velice obdobný komunikaci s WWW serverem. Dnes je SŘBD nejčastěji nepřetržitě spuštěn jako démon nebo jako služba a na určitém socketu očekává požadavky klientů (ostatních aplikací). Na tyto požadavky pak odpovídá. Vidíme tedy, že i zde funguje osvědčený model klient/server. V roli serveru je nyní SŘBD a někdy se mu proto také říká databázový server. Pro zadávání požadavků na databázový server aplikace nejčastěji používají jazyk SQL (Structured Query Language). Tento jazyk prošel dlouhým vývojem a v různé míře jej dnes podporují téměř všechny běžně používané databázové servery. Někdy se proto databázovým serverům říká zjednodušeně SQL servery. Jazyk SQL nabízí vše potřebné pro vytváření, modifikování a rušení tabulek a pro práci s údaji v tabulce — vyhledávání, přidávání, modifikování a mazání údajů. [1]

3.4.1 MySQL

Relační databázový server MySQL byl poprvé ukázán veřejnosti v roce 1996. Vyvinula ho švédská společnost MySQL AB. Software se stal velmi populární, a proto byl

zakoupen společností Oracle, která ho nyní vlastní. V dnešní době byl MySQL stažen více než sto milióny uživateli, mezi něž patří i velké společnosti a organizace jako Yahoo!, NASA, Google, a Cisco Systems. [16]

MySQL je multiplatformní, open source databázový server, který podporuje více než čtrnáct operačních systémů. Hlavní předností MySQL je rychlost, jelikož se na ni vývojáři zaměřili již od prvních verzí, a to i za cenu zjednodušení ostatních funkcí. [16]

3.4.2 PostgreSQL

PostgreSQL, původně nazvaný Postgres, byl vytvořen profesorem informatiky Michael Stonebrakerem. Dnes má za sebou již více než patnáct let aktivního vývoje. [17]

PostgreSQL je výkonný multiplatformní open source objektově-relační databázový systém. Je celosvětově uznáván jako nejspolehlivější a nejstabilnější databáze. Šance na zhroucení databáze jsou minimální a dokonce i v případě selhání databáze existují způsoby a funkce, které umožňují obnovit a zachránit tak data. [17]

3.5 UML – grafický modelovací jazyk

UML (Unified Modeling Language) je standardizovaný modelovací jazyk skládající se ze souboru diagramů vyvinutých pro systémové a softwarové vývojáře. Pomáhá vývojářům specifikovat, vizualizovat, vytvářet a dokumentovat artefakty softwarových systémů. UML představuje soubor nejlepších inženýrských postupů, které se osvědčily při modelování velkých a složitých systémů. UML je velmi důležitou součástí vývoje objektově orientovaného softwaru. Používá hlavně grafické notace k vyjádření návrhu softwarových projektů. [18]

3.5.1 Historie

Po rozšíření objektově orientovaného programování se hledal způsob, jak se dívat z různých pohledů na informační systémy. Vznikalo množství specifikací a standardů, jak objekty a další součásti aplikace zakreslovat. V průběhu 90. let se podařilo firmě Rational Software pomocí sjednocení několika metodik vytvořit standard UML. [19]

3.5.1.1 Konsorcium OMG

Časem se ukázalo, že velké korporace mají velký zájem na tom, aby UML standard existoval a byl nadále udržován. Proto vzniklo mezinárodní konsorcium OMG (Object Management Group), které dohlíží na specifikace UML. Jedná se o otevřenou instituci, kde figurují firmy jako IBM, Rational Software, Microsoft, Oracle, HP a další. Společně tak vytváří jazyk UML. [19]

Zatím poslední vydanou UML specifikací je verze 2.5.1. [20]

3.5.2 UML diagramy

UML v současné době definuje třináct diagramů, rozdělených do dvou základních kategorií.

- **Diagramy struktury (Structure Diagrams)** – popisují strukturu systému, tedy z čeho je složený.
- **Diagramy chování (Behaviour Diagrams)** – popisují chování systému, tedy jak funguje.

V diagramech chování nalezneme ještě samostatnou skupinu diagramů interakce.

- **Diagramy interakce (Interaction Diagrams)** – popisují interakci mezi jednotlivými částmi systému. [21]

3.5.2.1 Třídní diagram

Třídní diagram (Class Diagram) patří do skupiny diagramů struktury. Je to centrální modelovací technika, která prochází téměř všemi objektově orientovanými metodami. Tento diagram popisuje typy objektů v systému a různé druhy vztahů mezi nimi. [18]

Existují tři hlavní typy vztahů, které jsou důležité:

- Asociace – indikuje vztah mezi instancemi dvou a více tříd. Často popisuje schopnost jedné instance poslat zprávu druhé instanci.
- Kompozice – je nejsilnějším z trojice vztahů asociace – agregace – kompozice. Odkazovaný objekt je zásadní součástí majitele. Existence odkazovaného objektu bez majitele nemá smysl. Zánikem majitele tak většinou zaniká i odkazovaný objekt.
- Agregace – je specifitější vztah než asociace a volnější vztah než kompozice. Oproti kompozici je agregace volnější v tom, zánik majitele nemusí zapříčinit zánik odkazovaného objektu. [22]

3.5.2.2 Diagram případů užití

Diagram případů užití (Use Case Diagram) patří do skupiny diagramů chování. Účelem diagramu je popsat funkcionalitu systému, tedy co od něj očekáváme. Diagram vypovídá o tom, co má systém umět, ale neříká, jak to bude dělat. [19]

Use Case diagram se skládá z případů užití (Use Case), aktérů (Actors) a vztahů mezi nimi:

- Případ užití – je sada několika akcí, které vedou k dosažení určitého cíle. Definuje tedy jednu funkcionalitu, kterou by měl navrhovaný systém umět.
- Aktér – je role, která komunikuje s jednotlivými případy užití. V této roli může být obsazen uživatel nebo externí systém. [19]
- Vztahy – účast aktéra v případě použití se projevuje pevným spojením. [18]

3.6 QR kódy

QR (Quick Response) kódy jsou druh dvourozměrného čárového kódu, který slouží k rychlému přístupu k informacím. Nejčastěji k přenosu libovolné informace do chytrého mobilního telefonu či tabletu. [23]

QR kód je díky svému identifikačnímu vzorku navržen pro rychlé čtení pomocí optických snímačů CCD nebo CMOS, které jsou v dnešní době využívány ve fotoaparátech chytrých mobilních telefonů. [24]

3.6.1 Přístup k informacím

K informacím zakódovaným na QR kódu se může přistupovat pomocí mobilního zařízení vybaveného čtečkou QR kódů. Čtečka interpretuje data v kódu a přesměruje webový prohlížeč telefonu na cíl určený tvůrcem kódu. [25]

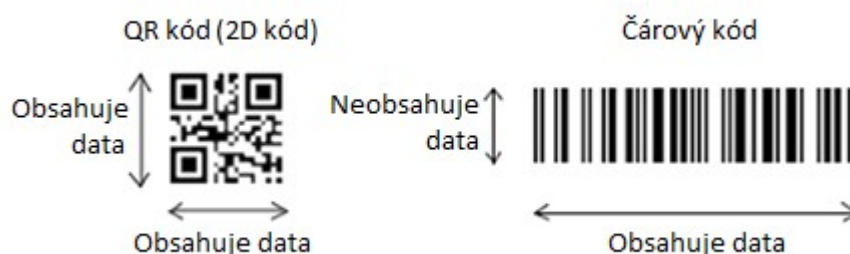
3.6.1.1 Čtečka QR kódů

Některá mobilní zařízení mají čtečku QR kódů již předinstalovanou. Ostatní vyžadují její stažení a nainstalování. [25]

Po spuštění čtečky QR kódů se pro načtení kódu použije fotoaparát mobilního zařízení a aplikace potom automaticky načte kódovaná data. [23]

3.6.2 Srovnání QR kódů s tradičními čárovými kódy

QR kódy pocházejí z devadesátých let, ale tradiční dvourozměrné čárové kódy existují již po desetiletí. Čárové kódy a QR kódy umožňují ukládat informace v malém prostoru. V běžném čárovém kódu se můžou kódovat data pouze vertikálně. V kódu QR se data kódují jak vertikálně, tak i horizontálně. [25]



Obrázek 1 - QR kód umožňuje ukládat více informací než čárový kód

QR kódy mají několik výhod:

- Jsou čitelné z libovolného směru.
- QR kódy jsou obecně trvanlivější. Informace v nich mohou být dekodovány, i když jejich část chybí nebo je poškozena.
- Hlavní hodnotou QR kódů je to, že rychle a snadno propojují offline svět s online obsahem. [25]

3.6.3 Další 2D formáty čárových kódů

QR kódy jsou formou 2D čárového kódu a jsou pravděpodobně nejčastější. Další známé formáty 2D čárových kódů zahrnují:

- Maticový 2D kód (DataMatrix) – byl navržen k uložení velkého množství informací na velmi malý prostor, je tedy snadno čitelný bez ohledu na svou velikost. Používá se především pro označování elektronických součástek (procesorů, čipů). [23]



Obrázek 2 - DataMatrix

- Aztécký kód – Byl vyvinut s důrazem na snadný tisk a jednoduché dekódování. Při tisku symbolu není třeba bílé ohraničení okolo symbolu. [24]

Používá se převážně v dopravě, na elektronických jízdenkách nebo palubních lístcích. [26]

V roce 2012 ho začaly používat s přechodem na termotisk České dráhy. [27]



Obrázek 3 - Aztécký kód

- PDF 417 – Kód s velmi vysokou informační kapacitou a schopností detekce a oprav chyb. Na rozdíl od tradičních čárových kódů, které obvykle slouží jako klíč k vyhledání údajů v nějaké databázi externího systému, si PDF 417 nese všechny údaje s sebou a stává se tak nezávislý na vnějším systému. [24]

Do PDF 417 lze zakódovat nejenom běžný text, ale i grafiku nebo speciální programovací instrukce. Příkladem použití mohou být nejrůznější identifikační karty nebo řidičské průkazy (v některých státech USA). [24]



Obrázek 4 - PDF 417

3.7 Mobilní platformy

Pojem mobilní platforma se užívá v souvislosti s chytrými mobilními telefony, tzv. smartphony. Platformy se vzájemně liší jak funkcemi poskytovanými operačním systémem uživatelům, tak i rozhraním operačního systému pro vývojáře mobilních aplikací a v neposlední řadě aplikacemi třetích stran, které jsou na dané platformě k dispozici. [28]

Nejperspektivnějšími mobilními platformami jsou dnes Google Android, Apple iOS a Microsoft Windows Phone. [28]

3.7.1 iOS

Operační systém iOS je mobilním systémem vytvářeným společností Apple, který vychází z desktopového operačního systému Mac OS X. [28]

V roce 2007 Apple oznámil iPhone, který značil začátek budování vlastního mobilního telefonu. Produkt zahrnoval řadu funkcí, jako je dotyková obrazovka a důraz na uživatelský zážitek, který se rychle stal standardem pro chytré mobilní telefony. V mnoha ohledech zůstává iPhone zlatým standardem pro mobilní telefony i dnes, přestože se trh rychle vyvíjí a inovuje. [29]

Mobilní operační systém iOS lze nalézt také na tabletu, zvaném iPad, stejně tak na iPodu, přenosném hudebním přehrávači. Protože společnost vyrábí jak zařízení, tak i operační systém, udržuje si vysokou úroveň kontroly nad svým ekosystémem. [29]

Aplikace pro iOS se vyvíjí v jazyku C a Objective-C. Na platformě Mac OS X je k dispozici vývojové prostředí od Apple XCode, nabízené zdarma. [28]

Každá aplikace umístěná v App Store musí projít schvalovacím procesem ze strany Apple, což má jak pozitivní tak negativní důsledky. Až do července roku 2008 nebyly aplikace třetích stran pro iOS oficiálně podporovány. [28]

3.7.2 Android

Operační systém Android je mobilním systémem vytvářeným společností Google, která ho zakoupila v roce 2005. První verze systému Android byla představena v roce 2008. Dnes Android s přehledem ovládá dlouhodobě trh se softwarem pro mobilní telefony. [29]

Jedná se o open source operační systém, což je výhodou pro výrobce mobilních telefonů, protože nemusí za jeho použití platit. To podstatně snižuje cenu telefonů vybavených tímto systémem. [28]

Operační systém Android lze nalézt také na zařízeních, jako jsou Google TV nebo Amazon Kindle Fire. [29]

Aplikace pro Android jsou distribuovány přes Google Play, provozovaný Googlem. V současnosti nabízí Google Play stovky tisíc aplikací, které si mohou uživatelé těchto telefonů stáhnout zdarma. [28]

Aplikace pro Android se vyvíjí s pomocí vývojového prostředí Android Studio s využitím programovacího jazyka Java a knihoven distribuovaných Googlem. [28]

3.7.3 Windows Phone

Operační systém Windows Phone je mobilním systémem vytvářeným společností Microsoft. [28]

V roce 2010 Microsoft vydal svoji moderní mobilní platformu Windows Phone 7, která nebyla zpětně kompatibilní s Windows mobile 6. Uživatelské rozhraní v systému Windows Phone 7, nazvané Metro, je zcela odlišné od přístupu, který využívají iOS i Android. Silný důraz je kladen na jednoduchost, typografii a uživatelsky atraktivní rozhraní. [29]

Distribuce aplikací probíhá přes Windows Phone Marketplace. Bohužel, více než rok po vzniku platformy Windows Phone 7 je stále velkým problémem naprostý nedostatek kvalitních aplikací pro tuto platformu. [28]

Pro vývoj aplikací můžeme použít vývojovou platformu Xamarin, která nám umožňuje psát mobilní aplikace v jazyce C#. [30]

4 Vlastní práce

Kapitola se věnuje postupu tvory zadaného softwaru. Na úvodu je popsána analýza a plánování činností, které je potřeba provést. Dále je navržen model celého systému, který zahrnuje diagram případů užití, model databáze a model objektů. Na závěr kapitoly je popsáno testování celé aplikace.

4.1 Analýza

Velmi důležitou součástí vytváření nové aplikace či systému je analýza všech požadavků a funkcionalit, které by měla aplikace splňovat. Slouží k přípravě veškerých podkladů potřebných pro realizaci. Z těchto požadavků pak vychází celý návrh a implementace. Důležité také je naplánovat průběh celého vývoje a rozložit větší problémy na podproblémy.

4.1.1 Plánované činnosti

Činnosti, ze kterých bylo vycházeno při tvorbě aplikace:

- Analýza veškerých funkcionalit a požadavků
- Návrh systému aplikace
- Implementace aplikace
- Testování

4.1.2 Požadavky na aplikaci

- Vyhledání místnosti v budově pomocí jejího interního čísla
- Vyhledání osob v kancelářích budovy
- Vyhledání pracoviště
- Multijazyčnost aplikace

4.1.3 Výběr vhodné mobilní platformy

Dnes patří mezi tři nejpoužívanější mobilní platformy Android, iOS a Windows Phone. Platforma Windows Phone, která ovládá přibližně jedno procento trhu, není vhodnou volbou. Android a iOS dohromady pokrývají přes devadesát pět procent trhu. Platforma Android je vhodnější, jelikož pokrývá sedmdesát pět procent trhu, vyvíjet pro ni lze na většině operačních systémech a publikování do aplikačního obchodu Google Play je snazší. Vývoj pro iOS platformu vyžaduje zařízení od firmy Apple s operačním

systemem macOS a App Store vyžaduje k uveřejnění aplikace daleko vyšší standardy než Google Play pro Android.

Na základě co nejjednodušší implementace a pokrytí nejvíce uživatelů byl projekt realizován jako webová aplikace, kde nezáleží na platformě mobilního zařízení uživatele. Jediné, co uživatel potřebuje vlastnit, je čtečka QR kódů, která dokáže výsledek zobrazit v kterémkoli webovém prohlížeči.

4.2 Modelový návrh systému

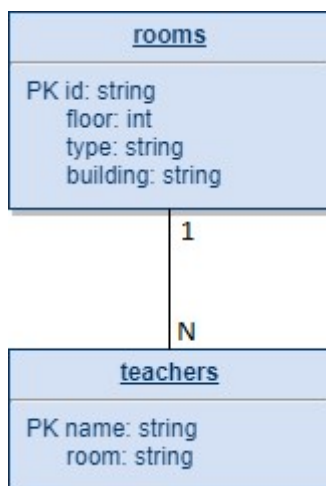
4.2.1 Databázový model

Databáze je navržena podle analýzy požadavků na aplikaci. Výsledky jsou zobrazeny v následujícím ER (Entity Relationship) diagramu. Diagram přesně kopíruje entity z reálného světa, které se v systému nachází.

Výsledná databáze aplikace při implementaci pak bude vycházet právě z tohoto modelu.

Tabulky nacházející se v ER diagramu:

- Rooms – Místnosti fakulty PEF.
- Teachers – Osoby v kancelářích fakulty PEF.



Obrázek 5 - Datový model

4.2.2 Diagram případů užití

Výsledky analýzy funkcionalit, které požadujeme od aplikace, jsou zahrnuty v diagramu případů užití (Use Case Diagram). Jsou zde zobrazeny osoby, které s aplikací přijdou do styku a činnosti jimi prováděné.

Výsledný diagram je přiložen jako příloha 1 této práce a podle něj se implementují prvky splňující požadované funkce.

4.2.3 Objektový model

Model je navržen podle analýzy požadavků na aplikaci a to stejně jako u databázového modelu. Entity vyskytující se v systému jsou v modelu reprezentovány samostatnými objekty. Tyto systémové objekty jsou navrženy tak, aby co nejlépe kopírovali skutečné entity a jejich chování.

4.3 Návrh vizuální reprezentace

4.3.1 Návrh UI

Návrh kvalitního a funkčního uživatelského rozhraní webové aplikace není jednoduchou záležitostí. Zdálo by se, že návrh uživatelského rozhraní je pouze otázkou vizuálního vzhledu a také rozmístění funkčních prvků na stránce. Avšak pokud má být webová aplikace úspěšná, je potřeba složitější pohled na celý proces vývoje.

Nejdůležitější prvky při návrhu uživatelského rozhraní jsou funkčnost a přínos celé aplikace, efektivita, jednoduché a intuitivní rozhraní a jednoduchost zapamatování.

Výsledný grafický vzhled celé webové aplikace je dotvořen pomocí technologie CSS. A výsledek, jak by mohla aplikace vypadat na chytrém mobilním telefonu je zobrazen v příloze 2 této práce.

4.3.1.1 Důležité fáze procesu tvorby UI

- Strategie
- Specifikace požadavků
- Návrh struktury
- Základní model a kostra
- Vizuální vzhled
- Testování

4.3.2 Návrh QR kódu

Na internetu existuje mnoho webových aplikací pro tvorbu QR kódů. Ne všechny umožňují generování kódu s logem či grafikou a některé jsou placené.

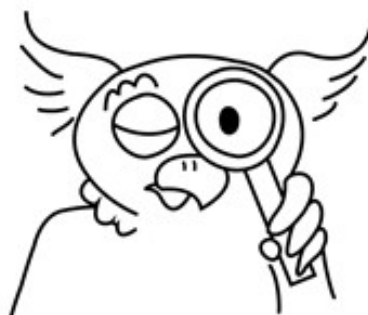
Jelikož námi požadované QR kódy mají podle specifikace obsahovat vloženou grafiku, byla zvolena pro jejich generování a následnou úpravu webová aplikace en.qrcode-pro.com.

Vygenerovaný QR kód, který by mohl být umístěn u schodiště v druhém patře ve hlavní budově fakulty PEF lze nalézt jako přílohu 3 této práce.

4.3.3 Návrh loga

Předlohou loga, které bude vkládané do QR kódů, bylo použito logo fakulty PEF, které znázorňuje sovu. Vzniklo tedy originální logo se sovou, která drží lupu, čímž znázorňuje hledání a mělo by být jasné, že slouží k orientaci.

Na výrobu loga byl použit open source bitmapový grafický editor GIMP 2.



Obrázek 6 - Vytvořené logo

4.4 Použité technologie

4.4.1 Programovací jazyk

Jako programovací jazyk byl zvolen jazyk PHP v nejnovější verzi 7.0, jelikož jde o jazyk multiplatformní, jednoduchý a přitom dostatečně robustní na všechny požadované úkony.

Jako hostingový server byl použit server php5.cz, který vznikl pro podporu uživatelů, programátorů a vývojářů, kteří se chtějí seznámit a v ostrém provozu vyzkoušet nejnovější technologii PHP.

4.4.2 Operační systém

Vzhledem k tomu, že je programovací jazyk PHP multiplatformní. Může být aplikace nasazena jak na kombinaci operačního systému Windows s IIS serverem tak na operačním systému Linux s Apache serverem.

Na hostovacím serveru php5.cz, který byl vybrán k publikování výsledné aplikace, je používán Linux s Apache serverem.

4.4.3 Systém řízení báze dat

Na hostovacím serveru běží MySQL verze 5.7 jako systém řízení báze dat s databázovým úložištěm InnoDB. Tato databáze není nikterak limitující a naopak její výhodou je rychlé provádění transakcí v reálném čase. V budoucnu by InnoDB mohl nahradit až 20x rychlejší modul TokuDB jenž hostovací server php5.cz nabízí.

4.5 Implementace

4.5.1 Vývojové prostředí

Aplikace byla vytvářena na notebooku s operačním systémem od společnosti Microsoft Windows verze 10 ve vývojovém prostředí Visual Studio Code doplněným o plugin pro PHP IntelliSense.

4.5.2 Verzovací systém

Aplikaci bylo během vývoje potřeba zálohovat a udržovat informace o jednotlivých verzích. K tomuto účelu byl využit verzovací systém git, který stvořil Linus Torvalds v roce 2005 a v současnosti jde o nejpoužívanější verzovací systém používaný například společnostmi jako je Microsoft nebo Google.

Aplikace byla vytvářena v lokálním a tedy i privátním git repozitáři, který byl následně jako jeden commit veřejně nahrán na GitHub.

4.5.3 Výsledné nasazení

Webová aplikace byla během implementace postupně nahrávána na vybraný webový hosting server php5.cz, kde byla průběžně testována funkčnost a výsledná webová aplikace je zde dostupná. Společně s webovou aplikací se na hostingovém serveru nachází MySQL databáze. Přístup na tento webový hostingový server je realizován pomocí protokolu FTP.

4.6 Testování

Testování softwaru je empirický technický výzkum kvality testovaného produktu, v tomto případě webové aplikace.

4.6.1 Testovací scénáře

Předpoklad pro testovací scénáře: uživatel aplikace úspěšně naskenoval QR kód nacházející se v budově.

- Testovací scénář 1
 - Proces hledání místnosti ve stejné budově, ve stejné sekci budovy. Uživatel zadal identifikaci místnosti.
- Testovací scénář 2
 - Proces změny jazyka a následné vyhledání kterékoli místnosti. Výsledkem musí být stálé udržení zvoleného jazyka.
- Testovací scénář 3
 - Proces vyhledání osoby v kanceláři. Osoba se nachází v místnosti mimo budovu, ve které se nachází skenovaný QR kód.

Výsledky všech výše zmíněných testovacích scénářů dopadly pozitivně.

4.6.2 Testování zobrazení

Webová aplikace byla testována, zda ji všechny majoritní prohlížeče zobrazují správně. Testování úspěšně proběhlo jak na prohlížečích na počítačích tak i na mobilních zařízeních.

- Windows 10 a Chrome 65
- Windows 10 a Firefox 58
- Windows 10 a Edge 16
- iOS 11.2 a Safari 11
- Android 8.1 a Chrome 65
- Windows 10 Mobile a Edge

4.6.3 Mobilní platformy

Pro správné zobrazení webové aplikace z naskenovaného štítku s QR kódem je potřeba mobilní zařízení s čtečkou QR kódů. Je třeba otestovat, zda všechny tři hlavní mobilní platformy (Android, iOS, Windows Phone) takovou čtečku mají, nebo je možnost si QR čtečku stáhnout z aplikačního obchodu. Čtečka by měla bez problémů skenovat námi vygenerované QR kódy a automaticky nabízet otevření aplikaci bez chyb ve webovém prohlížeči.

5 Výsledky a diskuse

Základem jakékoli realizace softwaru je důsledná analýza a rozbor požadavků. Nejlepší cestou je rozdělit si vše na menší kousky a s těmi následně pracovat. V průběhu dalšího vývoje je pak k dispozici celý návrh a lze ho upravovat pouze v malých detailech.

V průběhu analýzy bylo zjištěno, že nejlepším řešením, které pokryje nejvíce uživatelů, není vývoj aplikace pro jednu konkrétní platformu, ale výběr multiplatformního řešení. Takové řešení mohlo být implementováno například přes platformu Xamarin, která umožňuje vývoj pro všechny tři hlavní mobilní operační systémy v rámci jednoho sdíleného kódu. Toto řešení je ale stále velmi pracné a vyžaduje schválení výsledné aplikace společnostmi vlastnícími aplikační obchody. Vybraná implementace webové aplikace pak splňuje nejjednodušší řešení na implementaci a největší pokrytí uživatelů, protože není závislá na mobilní platformě.

Webová aplikace pro orientaci v budově poskytuje uživatelům nový rozměr v přístupu k online datům. Díky technologii QR kódu se offline svět s online daty dá velmi snadno propojit. A proto orientace v budově nebyla nikdy snazší.

6 Závěr

V teoretické části této bakalářské práce se čtenář seznámil s problematikou týkající se webových služeb a jejich použití, jak spolu navzájem komunikují nezbytné části systému klient a server. Bylo představeno propojení mezi offline světem s online obsahem pomocí QR kódu, ke kterému je potřeba pouze QR čtečka. Na závěr teoretické části byly vypsaný dnes nejdůležitější a nejpoužívanější mobilní platformy po světě.

V praktické části byl čtenář zahrnut nezbytností prvotní analýzy systému. Následovanou přímo návrhem aplikace, který kopíruje standardy UML a dalších norem softwarového inženýrství. V závěru této kapitoly je popsáno nezbytné testování aplikace před konečným použitím.

Výsledek, kterým je webová aplikace vytvořená v programovacím skriptovacím jazyce PHP, byl nasazen na veřejný server.

7 Seznam použitých zdrojů

1. KOSEK, Jiří. *PHP - tvorba interaktivních internetových aplikací: podrobný průvodce*. Praha: Grada, 1998. Průvodce (Grada). ISBN 80-716-9373-1.
2. *Studijní materiál k Tvorbě webových stránek* [online]. [cit. 2018-03-11]. Dostupné z: <http://jikos.cz/~perm/html/snas.html>
3. KOSEK, Jiří. *Vše o WWW* [online]. 1999 [cit. 2018-03-11]. Dostupné z: <http://www.kosek.cz/clanky/iweb/06.html>
4. *Webová Aplikace* [online]. 2016 [cit. 2018-03-11]. Dostupné z: <https://managementmania.com/cs/webova-aplikace-web-application>
5. DEMEL, Jiří. *Klient, server a protokol* [online]. 1990 [cit. 2018-03-11]. Dostupné z: <https://kix.fsv.cvut.cz/~demel/i4z/node6.htm>
6. *Internetové protokoly* [online]. 2007 [cit. 2018-03-11]. Dostupné z: <http://www.gvp.cz/local/new/ucebnice/Vyptech/internet/protokoly.htm>
7. GRYGAREK, Petr. *Základní vlastnosti protokolu HTTP* [online]. [cit. 2018-03-11]. Dostupné z: <http://www.cs.vsb.cz/grygarek/kotasek/htp02.htm>
8. PROCHÁZKA, David. *CSS a XHTML: tvorba dokonalých WWW stránek krok za krokem*. 2., aktualiz. vyd. Praha: Grada, 2011. Průvodce (Grada). ISBN 978-80-247-3897-0.
9. PROCHÁZKA, David. *Hledáme na internetu v rekordním čase*. 2., aktualiz. vyd. Praha: Grada, 2007. V rekordním čase. ISBN ISBN978-80-247-1471-4.
10. DUCKETT, Jon. *HTML & CSS: design and build websites*. Indianapolis, IN: Wiley, 2011. ISBN ISBN978-1-118-00818-8.
11. *IIS vs Apache* [online]. 2017 [cit. 2018-03-11]. Dostupné z: <https://www.upguard.com/articles/iis-apache>
12. *HTML 5.2* [online]. 2017 [cit. 2018-03-11]. Dostupné z: <https://www.w3.org/TR/html52/>
13. MARTIN, Michálek. *Průvodce CSS preprocesory* [online]. 2014 [cit. 2018-03-11]. Dostupné z: <https://www.vzhurudolu.cz/blog/12-css-preprocesory-1>
14. *What is PHP?* [online]. [cit. 2018-03-11]. Dostupné z: <http://php.net/manual/en/intro-what-is.php>
15. *PHP releases* [online]. 2018 [cit. 2018-03-11]. Dostupné z: <http://php.net/releases/>

16. GILMORE, W. J. *Velká kniha PHP 5 a MySQL: kompendium znalostí pro začátečníky i profesionály*. Nové, 3. vyd. Brno: Zoner Press, 2011. Encyklopedie Zoner Press. ISBN 978-80-7413-163-9.
17. *About PostgreSQL* [online]. [cit. 2018-03-11]. Dostupné z: <https://www.postgresql.org/about/>
18. *What is Unified Modeling Language?* [online]. [cit. 2018-03-11]. Dostupné z: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/>
19. DAVID, Čápka. *Úvod do UML* [online]. 2013 [cit. 2018-03-11]. Dostupné z: <https://www.itnetwork.cz/navrh/uml/uml-uvod-historie-vyznam-a-diagramy/>
20. *About the Unified Modeling Language specification version 2.5.1* [online]. [cit. 2018-03-11]. Dostupné z: <https://www.omg.org/spec/UML/About-UML/>
21. *Introduction to OMG'S Unified Modeling Language* [online]. 2005 [cit. 2018-03-11]. Dostupné z: <http://www.uml.org/what-is-uml.htm>
22. MILOŠ, Němec. *UML: Diagramy tříd* [online]. [cit. 2018-03-11]. Dostupné z: <http://www.milosnemec.cz/clanek.php?id=199>
23. *What is a QR Code?* [online]. 2013 [cit. 2018-03-11]. Dostupné z: <http://www.whatisaqrcode.co.uk/>
24. *Typy čárových kódů 2D* [online]. [cit. 2018-03-11]. Dostupné z: <http://www.duben.org/skola/fel/5.rocnik/NM/TypyKodu2D.htm>
25. WATERS, Joe. *QR codes for dummies*. Portable ed. Hoboken, NJ, 2012. --For dummies. ISBN 11-183-3703-4.
26. NIKOLOW, Peter. *Aztec code* [online]. 2012 [cit. 2018-03-11]. Dostupné z: <http://www.mobiliodevelopment.com/aztec-code-2d-barcode/>
27. *Projekt 2D kódů* [online]. 2013 [cit. 2018-03-11]. Dostupné z: <http://www.ceskedrahy.cz/tiskove-centrum/tiskove-zpravy/-16146/>
28. *Informace o mobilních platformách* [online]. 2013 [cit. 2018-03-11]. Dostupné z: <http://www.sunmarketing.cz/mobilni-aplikace/informace-o-mobilnich-platformach>
29. SHACKLES, Greg. *Mobile development with C#*. Sebastopol, CA: O'Reilly, c2012. ISBN 14-493-2023-6.
30. HERMES, Dan. *Xamarin Mobile Application Development: Cross-Platform C# and Xamarin.Forms Fundamentals*. California: Apress, 2015. ISBN 978-1-4842-0215-9.

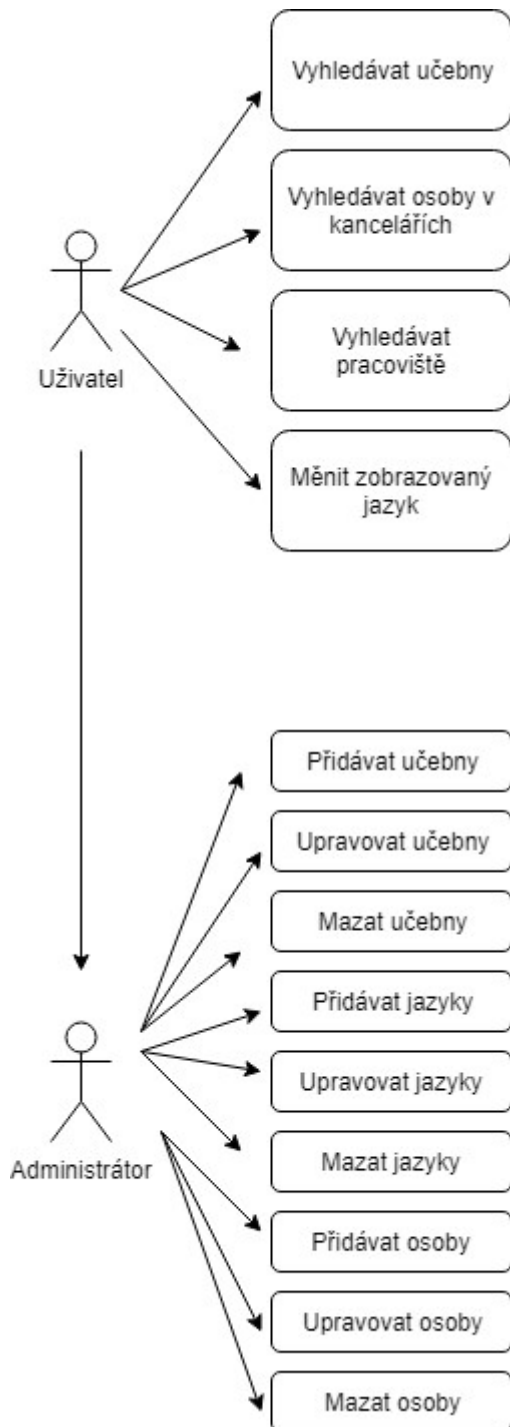
8 Přílohy

Příloha 1 – Use Case Diagram

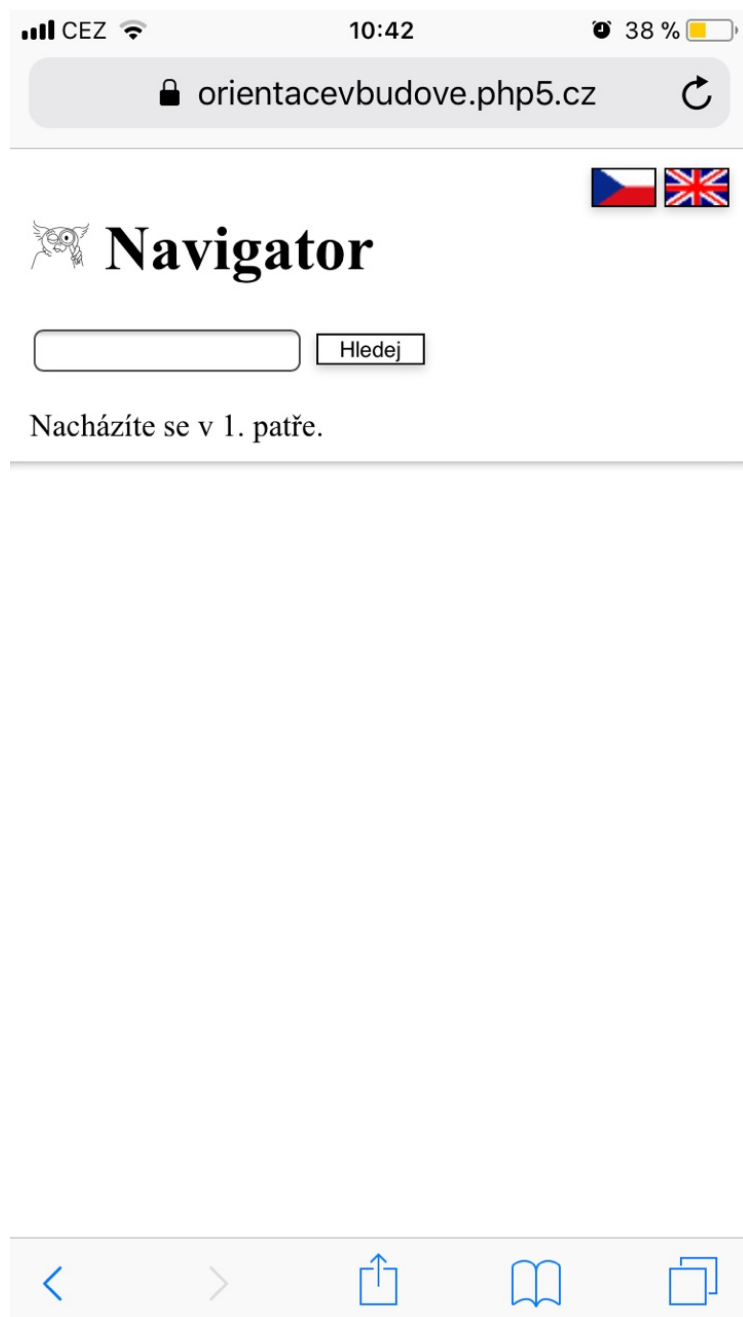
Příloha 2 – Uživatelské rozhraní

Příloha 3 – Vygenerovaný QR kód

Příloha 1 – Use Case Diagram



Příloha 2 – Uživatelské rozhraní



Příloha 3 – Vygenerovaný QR kód

