

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství



Bakalářská práce

Návrh a implementace webové aplikace pro portál zvířat

Lucie Zvelebilová

© 2021 ČZU v Praze

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Lucie Zvelebilová

Systémové inženýrství a informatika
Informatika

Název práce

Návrh a implementace webové aplikace pro portál zvířat

Název anglicky

Design and Implementation of Web Application for Animal Portal

Cíle práce

Cílem bakalářské práce je navrhnout a implementovat webovou aplikaci pro portál zvířat, který bude umožňovat registrovaným uživatelům adoptovat či inzerovat zvířata hledající domov.

Metodika

1. Na základě studia odborných zdrojů popište formou literární rešerše doménu vývoje webových aplikací dle osvědčených metod softwarového návrhu
2. Proveďte analýzu uživatelských požadavků na webovou aplikaci a popište je vhodnými nástroji (Use Case)
3. V souladu s doporučenými postupy softwarového inženýrství vytvořte návrh vlastní webové aplikace (konceptuální model)
4. Návrh aplikace implementujte ve vhodném prostředí a řádně otestujte

Doporučený rozsah práce

30-40 stran

Klíčová slova

webová aplikace, portál zvířat, informační systém

Doporučené zdroje informací

Fowler, S., & Stanwick, V. (2004). Web Application Design Handbook: Best Practices for Web-Based Software. San Francisco: Morgan Kaufmann Publishers.

Jablonski, S., Mayer, U., Petrov, I., & Meiler, C. (2004). Guide To Web Application And Platform Architectures. Berlin: Springer, Berlin, Heidelberg.

NIXON, Robin. Learning PHP, MySQL & JavaScript: with jQuery, CSS & HTML5. 4th. Beijing: O'Reilly Media, 2014.

Shklar, L., & Rosen, R. (2009). Web Application Architecture: Principles, Protocols and Practices (2nd). Hoboken: John Wiley.

Předběžný termín obhajoby

2019/20 LS – PEF

Vedoucí práce

Ing. David Buchtela, Ph.D.

Garantující pracoviště

Katedra informačního inženýrství

Elektronicky schváleno dne 14. 3. 2020

Ing. Martin Pelikán, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 14. 3. 2020

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 15. 03. 2021

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci " Návrh a implementace webové aplikace pro portál zvířat " 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 15.3.2021

Poděkování

Ráda bych touto cestou poděkovala Ing. Davidu Buchtelovi, Ph.D. za konzultace mé bakalářské práce a vhodné připomínky pro její zpracování.

Návrh a implementace webové aplikace pro portál zvířat

Abstrakt

Bakalářská práce se zabývá problematikou evidence opuštěných či ztracených zvířat. Řešením je návrh webové aplikace pro portál zvířat, kde lze zvířata inzerovat i adoptovat. Hlavní myšlenkou této aplikace je pomoci zvířatům nalézt nový domov, a proto její určení není vyhrazeno konkrétním uživatelům a portál mohou využívat i útulky pro zvířata či jiná zařízení, a tak zefektivnit propojení umístěných zvířat s novými či původními majiteli. Účelem této aplikace bude tedy zprostředkování centrální evidence ztracených a opuštěných zvířat za účelem usnadnění celého procesu adopce či nalezení ztraceného mazlíčka.

Klíčová slova: webová aplikace, portál zvířat, zvířecí útulek

Design and implementation of web application for animal portal

Abstract

The bachelor thesis deals with the issue of registration of abandoned or lost animals. The solution is to design a web application for the animal portal, where animals can be advertised and adopted. The main idea of this application is to help animals find a new home, and therefore its purpose is not reserved for specific users and the portal can be also used by animal shelters or other facilities, thus streamlining the connection of placed animals with new or original owners. The purpose of this application will be to provide a central register of lost and abandoned animals in order to facilitate the entire process of adoption or finding a lost pet.

Keywords: web application, animal portal, animal shelter

Obsah

1 Úvod.....	8
2 Cíl práce a metodika	9
Cíl práce	9
2.1 Metodika	9
2.2 Teoretická východiska	10
2.2.1 Útulek pro zvířata	10
2.2.1.1 Propojení útulků s portálem.....	10
2.2.1.2 Výhody užívání portálu	10
2.2.1.3 Nevýhody využití portálu	10
2.2.2 Webová aplikace.....	10
2.2.3 HTML (Hypertext Markup Language).....	10
2.2.4 CSS (Cascading Style Sheets)	11
2.2.5 PHP (Hypertext Preprocessor).....	11
2.2.6 MySQL (Structured Query Language)	11
2.2.7 Webový server	11
2.2.8 Relační databáze	11
2.2.9 Xampp.....	12
2.2.10 PhpMyAdmin.....	12
2.2.11 Notepad++	12
2.2.12 Wireframe	12
2.2.13 Informační systém.....	12
2.3 Metodické přístupy k tvorbě IS/ICT	12
2.3.1 Modely životního cyklu	12
2.3.1.1 Vodopádový přístup	13
2.3.1.2 Iterativní přístup	13
2.3.2 Etapy životního cyklu vývoje softwaru	13
2.4 Metodiky budování IS/ICT	13
2.4.1 Kategorizace metodik budování IS/ICT	14
2.4.1.1 Zaměření metodiky.....	14
2.4.1.2 Rozsah metodiky	14
2.4.1.3 Váha metodiky.....	14
2.4.1.4 Typ řešení	14
2.4.1.5 Doména.....	14
2.4.1.6 Přístup k řešení	14

2.4.2	RUP (Rational Unified Process)	15
2.4.3	Metodika Microsoft Solutions Framework.....	15
2.4.4	Agilní metodiky	15
2.4.4.1	Extremní programování.....	15
2.4.4.2	Scrum.....	15
Vlastní práce.....		16
2.5	Návrh vlastního řešení.....	16
2.5.1	Vzorové osoby	16
2.5.2	Vymezení funkcionalit.....	17
2.5.2.1	Základní přehled.....	17
2.5.2.2	Registrace a přihlášení.....	17
2.5.2.3	Přidávání inzerátů.....	17
2.5.2.4	Editace inzerátů	17
2.5.3	Logický design – Wireframe	17
2.5.3.1	Domů – úvodní stránka	17
2.5.3.2	K adopci.....	19
2.5.3.3	Přihlášení	20
2.5.3.4	Registrace	21
2.5.3.5	Moje inzeráty	22
2.5.3.6	Vložit inzerát	22
2.5.3.7	Kontakt	23
2.6	Metodika řešení.....	25
2.6.1	Xampp - instalace	25
2.6.2	Síťový port.....	25
2.6.3	Apache	25
2.6.4	MySQL	26
2.7	Implementace	26
2.7.1	Návrh databáze	26
2.7.1.1	ERD – Diagram Vztahu Entit.....	26
2.7.2	Vytvoření tabulky	27
2.7.2.1	Datové typy	27
2.7.2.2	Úložné databázové stroje (storage engine).....	27
2.7.3	Připojení k databázi	28
2.7.4	Filtr zvířat k adopci.....	28
2.7.5	Ukázkový kód pro vložení inzerátu	34
Závěr		40

3 Seznam použitých zdrojů	41
4 Přílohy	42
4.1 Příloha A Vytvořená aplikace – úvodní strana	43
4.2 Příloha B Vytvořená aplikace – filtr zobrazení zvířat k adopci	44
4.3 Příloha C Vytvořená aplikace – vložení inzerátu.....	44

Seznam obrázků

Obrázek 1 - Vodopádový model (10)	13
Obrázek 2 - Wireframe úvodní stránky	18
Obrázek 3 - Wireframe úvodní stránky (po přihlášení).....	18
Obrázek 4 - Wireframe pro zobrazení zvířat k adopci.....	19
Obrázek 5 - Wireframe pro zobrazení zvířat k adopci (po přihlášení)	20
Obrázek 6 - Wireframe pro přihlášení do portálu.....	21
Obrázek 7 - Wireframe pro vytváření nového účtu	21
Obrázek 8 - Wireframe pro vložení inzerce přihlášeného uživatele	22
Obrázek 9 - Wireframe pro nahrání inzerátu	23
Obrázek 10 - Wireframe pro stránku s kontaktem.....	24
Obrázek 11 - Wireframe pro stránku s kontaktem (po přihlášení)	24
Obrázek 12 - Diagram vztahu entit.....	26
Obrázek 13 – Symboly vztahů entit (13)	27

Seznam tabulek

Tabulka 1 - Vzorová persona (9)	16
Tabulka 2 - Vzorová persona (2)	16
Tabulka 3 - Vzorová persona (3)	16
Tabulka 4 - Vzorová persona (4)	16
Tabulka 5 –Síťové porty (12)	25

Seznam zdrojových kódů

Zdrojový kód 1 – Příkaz SQL pro vytvoření tabulky „inzeraty“	28
Zdrojový kód 2 - Příkaz SQL pro vytvoření tabulky „uzivatele“	28
Zdrojový kód 3 - Připojení k databázi MySQL v PHP	28
Zdrojový kód 4 – Adopce	31
Zdrojový kód 5 - Najdi zvíře	34
Zdrojový kód 6 - Vložení inzerátu.....	39

1 Úvod

V souvislosti s mým zájmem o zvířata se téma mé bakalářské práce týká návrhu a implementace webové aplikace pro portál zvířat za účelem pomoci zvířatům nalézt nový domov.

Využití této aplikace je vhodné i pro zvířecí útulky, které mohou prostřednictvím tohoto portálu inzerovat zvířata umístěna v těchto zařízeních. Lze tedy říct, že aplikace je použitelná nejenom pro opuštěná, ale i ztracená zvířata a uživatelé mohou prostřednictvím tohoto portálu snadněji najít správného mazlíčka.

Tato práce se dělí na dvě části, teoretickou a praktickou.

V teoretické části jsou stručně popsána teoretická východiska zahrnující popis technologií, nezbytných pro vývoj webové aplikace, metodické postupy při vytváření informačního systému a stručné pojednání o útulcích pro zvířata a následné srovnání využití tohoto portálu.

V praktické části je zobrazen návrh řešení webové aplikace, který také zahrnuje vzorové persony, popis funkcionalit portálu a logický design, tvořený pomocí wireframů.

Další součástí praktické části mé práce je také popis metodik řešení a implementace webové aplikace.

2 Cíl práce a metodika

Cíl práce

Cílem této bakalářské práce je navrhnout a implementovat webovou aplikaci pro portál zvířat, který bude umožňovat uživatelům adoptovat či inzerovat zvířata hledající domov. Portál bude zahrnovat zobrazení evidence zvířat k adopci, kde bude umožněno nastavit si kritéria pomocí filtru pro snadnější vyhledávání.

Využití portálu je cíleno také pro zvířecí útulky, které mohou prostřednictvím této aplikace inzerovat zvířata, která jsou tam umístěna. Hlavní myšlenkou této aplikace je vytvoření centrální databáze evidence opuštěných zvířat po celé ČR, a tak zajistit snadnější vyhledávání vhodného mazlíčka pro uživatele. Uživatelé díky tomuto portálu nebudou muset využít několik stránek s inzeráty, nýbrž pouze jednu.

2.1 Metodika

V rámci mé práce budu navrhovat a implementovat webovou aplikaci pro portál zvířat. Pro vytvoření aplikace využiji technologie pro vývoj webových aplikací a osvědčené postupy softwarového návrhu, které budu více zmiňovat v teoretické části. Praktická část bude především obsahovat logický návrh aplikace, vzorové osoby, ukázkové bloky zdrojového kódu a znázorněnou datovou strukturu.

2.2 Teoretická východiska

2.2.1 Útulěk pro zvířata

Definice útulku podle veterinárního zákona: útulek (§ 3 odst. 1 písm. kk)) – „útulkem pro zvířata se rozumí zařízení, které poskytuje dočasnou péči toulavým a opuštěným zvířatům. Útulěk je tedy místo, kde jsou umístěna zvířata, u kterých nelze identifikovat majitel a je to zařízení, které těmto zvířatům poskytuje dočasnou péči a domov. (1)

2.2.1.1 Propojení útulků s portálem

Jedno z primárních využití webové aplikace pro portál zvířat je možnost propojení útulků s touto aplikací. Prostřednictvím portálu je možné nahrávat opuštěná či zatoulaná zvířata na web, kde se jednodušeji nalezne majitel nový či stávající, pomocí filtrů kategorií, kde si uživatel může určit druh zvířete, pohlaví, stáří a kraj, kde se zvíře nachází. Primárním cílem je tedy sjednocení evidencí jakýchkoliv opuštěných zvířat v útulcích či volně žijících v ČR a vytvořit jednodušší způsob k nalezení vhodného mazlíčka pro každého uživatele.

2.2.1.2 Výhody užívání portálu

Výhodami užívání tohoto portálu pro útulky je snadnější a rychlejší způsob hledání vhodného zvířete pro uživatele, a to primárně díky centralizaci evidence zvířat v útulcích a možnosti vyhledávání dle kategorií, pomocí nastavení filtrů.

2.2.1.3 Nevýhody využití portálu

Mezi nevýhody portálu lze zařadit to, že součástí portálu není umístění pro zvířata, a tudíž není uživatelům umožněno se jet na zvířata podívat na určité místo. Nicméně s ohledem na tuto skutečnost se nabízí možnost využít návštěvu útulku nebo se domluvit s uživateli, kteří zvířata inzerují.

2.2.2 Webová aplikace

Webovou aplikaci lze definovat jako jakýsi program, který je možné spustit prostřednictvím webového prohlížeče na libovolném zařízení bez vyžadované instalace. Je to aplikace typu server / klient, která používá webový prohlížeč jako svůj klientský program a provádí interaktivní službu připojením k serverům přes internet (nebo intranet). Web dodává obsah ze statických souborů a Webová aplikace představuje dynamicky přizpůsobený obsah založený na parametrech požadavku, sledovaném chování uživatelů a bezpečnostních aspektech.

Hlavním rozdílem mezi webovou aplikací a webovou stránku je, že aplikace má přístup k databázi, prostřednictvím které může ukládat a načítat data. Zpravidla bývá složitější. (2)

2.2.3 HTML (Hypertext Markup Language)

HTML je značkovací programovací jazyk pro tvorbu webových stránek. Je jedním z původních základních kamenů webu, jehož účelem bylo umožnit křížové odkazy na dokumenty prostřednictvím hypertextových odkazů. (2)

2.2.4 CSS (Cascading Style Sheets)

CSS je jazyk pro nastavování stylů (např. barvy, písma), který se používá pro úpravu vzhledu webových stránek a jejich obsahu. Už z názvu vyplývá, že CSS je vlastně taková šablona stylu. Ta je tvořena pravidly, která určují, jak mají být zobrazovány elementy v HTML, XHTML nebo XML dokumentech. Existuje více způsobů, jak propojit tato pravidla stylu s dokumenty HTML. Nejvhodnější pro tvorbu webu je použití externího CSS, protože umožňuje změnit vzhled celého webu změnou pouze jednoho souboru. (2)

2.2.5 PHP (Hypertext Preprocessor)

PHP je skriptovací programovací jazyk, který je především určený k programování dynamických webových stránek a webových aplikací. V jazyce PHP na rozdíl od JavaScriptu a HTML jsou zdrojové kódy webových stránek uloženy na webovém serveru. PHP skript se provede na serveru a prohlížeči je následně zaslán pouze výsledek. (2)

2.2.6 MySQL (Structured Query Language)

MySQL je jeden z nejpoužívanějších systémů správy relačních databází. Je založen na jazyce SQL, který je užíván ke konfiguraci dat v databázi pomocí příkazů. Tento systém lze použít pro různé aplikace, ale nejčastěji se vyskytuje na webových serverech. Web, který používá MySQL, může zahrnovat webové stránky, které přistupují k informacím z databáze. Tyto stránky jsou často označovány jako „dynamické“, což znamená, že obsah každé stránky je generován z databáze při načítání stránky. (3)

2.2.7 Webový server

Webový server lze chápat jako počítač, který vyřizuje požadavky od klientů. Webový server umožňuje HTTP přístup k „webové stránce“, což je jednoduše soubor dokumentů a dalších informací, podobně jako systém souborů v počítači. Kromě poskytování přístupu ke statickým dokumentům implementují moderní webové servery řadu protokolů pro předávání požadavků na vlastní softwarové aplikace, které poskytují přístup k dynamickému obsahu. (3)

2.2.8 Relační databáze

Základem relačních databází jsou databázové tabulky (entity), mezi kterými existuje jistá logická vazba (relace). Sloupce v tabulce reprezentují vlastnosti (atributy) této entity a musí mít jedinečný název a určený datový typ (číslo, text, logická hodnota, atd.). Řádky tabulky reprezentují samotné záznamy s přiděleným jedinečným identifikátorem, podle kterého se určuje příslušný záznam (klíč). (4)

Mezi neznámější zástupce relačních databázových systémů patří MySQL, Oracle a další. Pro webové aplikace menšího rozsahu se nejčastěji používá MySQL ve spojení s PHP.(4)

2.2.9 Xampp

Xampp označuje multiplatformní softwarový balíček, který vyvinula firma Apache Friends a slouží pro vytváření lokálního webového serveru pro vývoj a testování.

Přestože na oficiálních stránkách Apache Friends není uvedeno přesné vysvětlení termínu Xampp, zřejmá zkratka by mohla být neoficiálně vysvětlena jako: „X“ = Cross-Platform, „A“ = Apache, „M“ = MariaDB, „P“ = PHP a „P“ = Perl. (5)

2.2.10 PhpMyAdmin

PhpMyAdmin je programový systém, který slouží pro editaci a správu obsahu databáze MySQL. Umožňuje vytvářet, upravovat či rušit databáze a tabulky. Dále provádět příkazy SQL a nastavování klíčů. (6)

2.2.11 Notepad++

Notepad++ je textový editor, který slouží pro zobrazení a úpravu zdrojového kódu. Je vhodnější než poznámkový blok, protože je rozšířen o podporu panelů pro snadnější editaci více souborů. (7)

2.2.12 Wireframe

Wireframe se vytváří jako logický návrh webu, který definuje rozmístění funkčních prvků a obsah webové stránky. Na rozdíl od grafického návrhu, wireframe je vytvářen pomocí čar a textu. (8)

2.2.13 Informační systém

Pojmem Informační systém se rozumí celek tvořený lidmi, softwarem a hardwarem. Mezi nimi probíhají určité procesy, které jsou vzájemně propojené a slouží k ukládání, zpracování a poskytování informací. (9)

2.3 Metodické přístupy k tvorbě IS/ICT

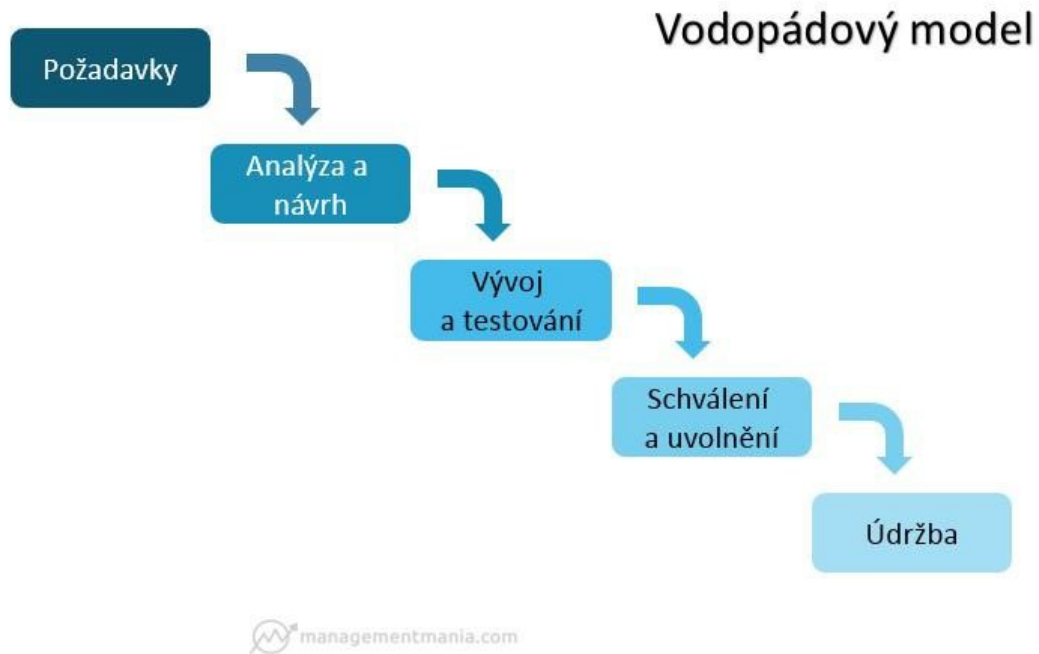
Metodické přístupy používáme pro zlepšení procesů tvorby IS a docílit jeho úspěšnosti. Ta může být definována 3 aspekty: dokončit projekt včas, podle rozpočtu a se všemi funkčními požadavky. (9)

2.3.1 Modely životního cyklu

Model životního cyklu je rámec procesů a aktivit organizovaných do fází. Existuje několik modelů tohoto cyklu, z nichž nejznámějšími jsou vodopádový, iterativní a inkrementální. (9)

2.3.1.1 Vodopádový přístup

Vodopádový přístup je tvořen jednotlivými etapami, které na sebe vzájemně navazují a jsou prováděny postupně. Název tohoto přístupu byl odvozen od jeho grafického znázornění, které připomíná vodopád. (9)



Obrázek 1 - Vodopádový model (10)

2.3.1.2 Iterativní přístup

Tento přístup odstraňuje nedostatky vodopádového vývoje. Rozkládá celý projekt na řadu dílčích projektů (iterací), protože zohledňuje skutečnost, že člověk daleko lépe řeší menší problémy. Výsledkem každé iterace musí být fungující část systému, která je otestována a integrována s výsledky té předchozí.

Iterativní model může být realizován dvěma základními modely životního cyklu – inkrementálním či evolučním. (9)

2.3.2 Etapy životního cyklu vývoje softwaru

- 1) Specifikace požadavků
- 2) Návrh
- 3) Implementace
- 4) Testování
- 5) Provoz a údržba

2.4 Metodiky budování IS/ICT

Metodiku lze obecně definovat jako souhrn postupů a úkolů. Postupy budování IS/ICT definují principy, techniky, procesy, role, nástroje a produkty vývoje informačního systému.

Tyto metodiky bývají také označovány jako postupy softwarového návrhu, ale tento název už je v dnešní době nedostačující, jelikož se ne vždy jedná o vytvoření návrhu nýbrž o nasazení hotového řešení či zlepšení toho stávajícího. V případě mé práce by se název užít mohl, protože se jedná o vytvoření nového návrhu software – webové aplikace. (9)

2.4.1 Kategorizace metodik budování IS/ICT

2.4.1.1 Zaměření metodiky

V rámci tohoto kritéria se rozhodujeme, zda se jedná o metodiku globální, která se týká celé organizace či projektovou metodiku, zabývající se pouze jedním určitým projektem. (9)

2.4.1.2 Rozsah metodiky

Za rozsah metodiky se považuje počet fází životního cyklu informačního systému. Na základě tohoto kritéria rozlišujeme metodiky pokrývající celý životní cyklus tvorby a dílčí metodiky. (9)

2.4.1.3 Váha metodiky

Pod váhu metodiky spadají charakteristiky označené zkratkou PARTS = precision (podrobnost), accuracy (přesnost), relevance (relevance), tolerance (tolerance), scale (měřítko) a od nich následně odvozená velikost, hustota a váha.

V této kategorii rozlišujeme těžké a lehké metodiky. (9)

2.4.1.4 Typ řešení

V dnešní době je používanější implementace aplikačního softwaru (Application Service Provision). Podle tohoto kritéria se rozlišuje vývoj nového řešení, integrace řešení, rozvoj a rozšíření řešení, implementace řešení a užití řešení. (9)

2.4.1.5 Doména

Toto kritérium se uplatňuje v případě projektových metodik. Představuje určitou oblast, pro kterou je IS vytvářen. (9)

2.4.1.6 Přístup k řešení

Zohledňuje paradigma, na kterém je metodika založena. Rozlišuje metodiky pro strukturovaný vývoj, rychlý vývoj aplikací (RAD), objektový vývoj, komponentový vývoj či vývoj služeb. (9)

2.4.2 RUP (Rational Unified Process)

Metodika RUP vznikla spojením přístupu Rational a metodiky Objectory Process a využívá se k vývoji softwaru. Proces vývoje může být popisován v rámci dvou dimenzí, při čemž jedna dimenze zohledňuje dynamický pohled na proces, vyjádřený pomocí cyklů, fází a iterací a druhá zohledňuje statické hledisko procesu, do kterého spadá popis činností, produktů, pracovníků atd. RUP je založena na iterativním vývoji a vychází z vizuálního modelování UML. Je složena ze čtyř fází (zahájení, projektování, realizace, předání). (9)

2.4.3 Metodika Microsoft Solutions Framework

Rámec pro vývoj softwaru, ve kterém je možné definovat metodiku pro konkrétní projekt. Rozlišuje dva modely – týmový a procesní. Týmový model se soustředí na pracovníky a jejich aktivity za účelem úspěšnosti projektu, zatímco procesní je iterativní model s krátkými iteracemi. (9)

2.4.4 Agilní metodiky

Agilní metodiky jsou především určeny pro vyvíjení softwaru založeném na iterativním a inkrementálním přístupu. Umožňují rychlý vývoj a dokáží reagovat na změny požadavků v průběhu vývoje. Tyto metodiky nepopisují procesy při vývoji softwaru, ale definují principy a praktiky. Mezi agilní metodiky se řadí např. Extrémní programování a metodika Scrum. (9)

2.4.4.1 Extrémní programování

Extrémní programování je metodika, která probíhá v krátkých cyklech, kde se navrhuje párové programování a jednotkové testování celého kódu. Základním principem této metodiky je dotažení osvědčených principů a postupů pro vývoj softwaru do extrému. V extrémním programování se vždy programuje pouze to, co je třeba ke splnění aktuálních požadavků. (9)

2.4.4.2 Scrum

Důležitou součástí této metodiky je každodenní krátké setkání týmu (4-10 lidí), kde jsou prezentovány jednotlivými členy jejich činnosti z předešlého dne, současné činnosti a plánované činnosti na budoucí dny. Technika Scrum využívá iterace zvané „Sprinty“, které mají určenou přesnou dobu trvání. Pokaždé po skončení této doby je prezentován postup vývoje zákazníkovi s očekávanou zpětnou vazbou. (9)

Vlastní práce

2.5 Návrh vlastního řešení

Návrhem řešení je webová aplikace, vytvořena na základě získaných informací tak, aby byla uživatelsky přívětivá. Při vytváření aplikace je nutno zohlednit funkcionalitu a logické rozdělení pro budoucí užívání. Dále je také důležité vytvořit vzorové osoby pro představu, komu je aplikace určena a kdo by jí měl užívat, jaké technologie jsme použili k jejímu vytvoření a grafické návrhy.

2.5.1 Vzorové osoby

Tabulka 1 - Vzorová osoba (9)

Jméno:	Arleta Urbanová
Pohlaví:	žena
Věk:	25
Zaměstnání:	student
Zájmy:	Cestování, Ekologie a životní prostředí, čtení, hudba
Vztah ke zvířatům:	Má velmi dobrý vztah ke zvířatům. Vlastní dva mazlíčky, kočku a psa, o které se velmi dobře stará. Je veganka a bojuje proti násilí na zvířatech.

Tabulka 2 - Vzorová osoba (2)

Jméno:	Richard Albrecht
Pohlaví:	muž
Věk:	49
Zaměstnání:	Učitel na SŠ
Zájmy:	Ornitologie, cestování, čtení,
Vztah ke zvířatům:	Richardův vztah ke zvířatům je kladný. Ačkoliv nejvíce upřednostňuje ptactvo, vlastní 2 psi a 1 kočku.

Tabulka 3 - Vzorová osoba (3)

Jméno:	Matěj Mochan
Pohlaví:	muž
Věk:	26
Zaměstnání:	kuchař
Zájmy:	vaření, hudba, čtení
Vztah ke zvířatům:	Matěj má jednoho psa, kterého se ujal, aby neskončil v útulku.

Tabulka 4 - Vzorová osoba (4)

Jméno:	Marie Turková
Pohlaví:	žena
Věk:	72
Zaměstnání:	V důchodu
Zájmy:	Zahradničení, luštění křížovek, vaření, turistika

Vztah ke zvířatům:	Marie má ráda zvířata. Žije sama, a proto by si ráda brzo nějaké pořídila.
--------------------	--

2.5.2 Vymezení funkcionalit

2.5.2.1 Základní přehled

Základním přehledem jsou stránky, které jsou dostupné pro uživatele bez přihlášení. Jedná se o domovskou stránku, která se uživateli zobrazí po spuštění webové aplikace, dále stránka s přehledem vložených inzerátů se zvířaty k adopci a kontakt.

2.5.2.2 Registrace a přihlášení

Součástí webové aplikace je registrace a přihlášení. Pro zpřístupnění všech funkcionalit pro uživatele, je zapotřebí se nejdříve registrovat a přihlásit. Bez úspěšného přihlášení není uživateli umožněno přidání inzerátu a nemá k dispozici evidenci inzerátů, které vložil.

2.5.2.3 Přidávání inzerátů

Přidání zvířete nejprve vyžaduje nahrání obrázku. Dalším krokem je napsat jméno zvířete, popř. jak zvíře oslovujete, pokud nemá přidělené jméno. Dále je nutné vybrat z možností, co je to za zvíře. Součástí je také určení pohlaví, stáří zvířete, alespoň přibližně a kraj, ve kterém se zvíře nachází.

2.5.2.4 Editace inzerátů

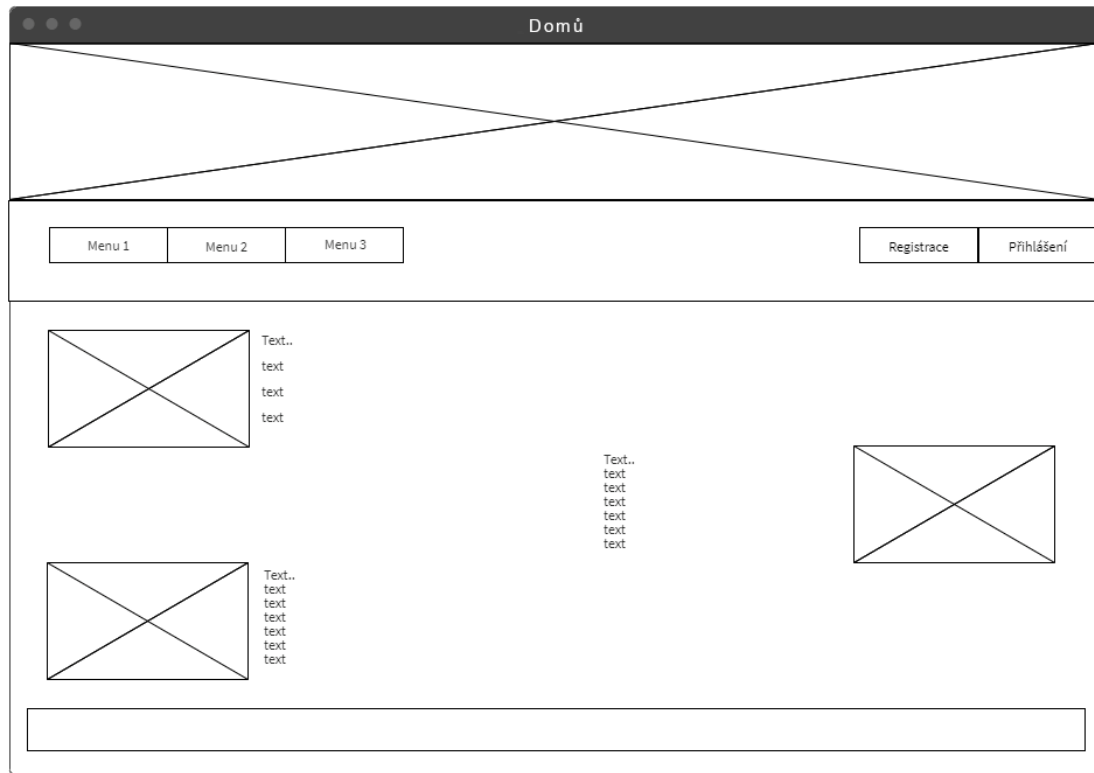
Editace inzerátu není umožněna uživateli a editaci zprostředkovává admin, který je uveden jako správce na stránce Kontakt, kde je k nalezení i způsob kontaktování pro editaci či odstranění inzerátu.

2.5.3 Logický design – Wireframe

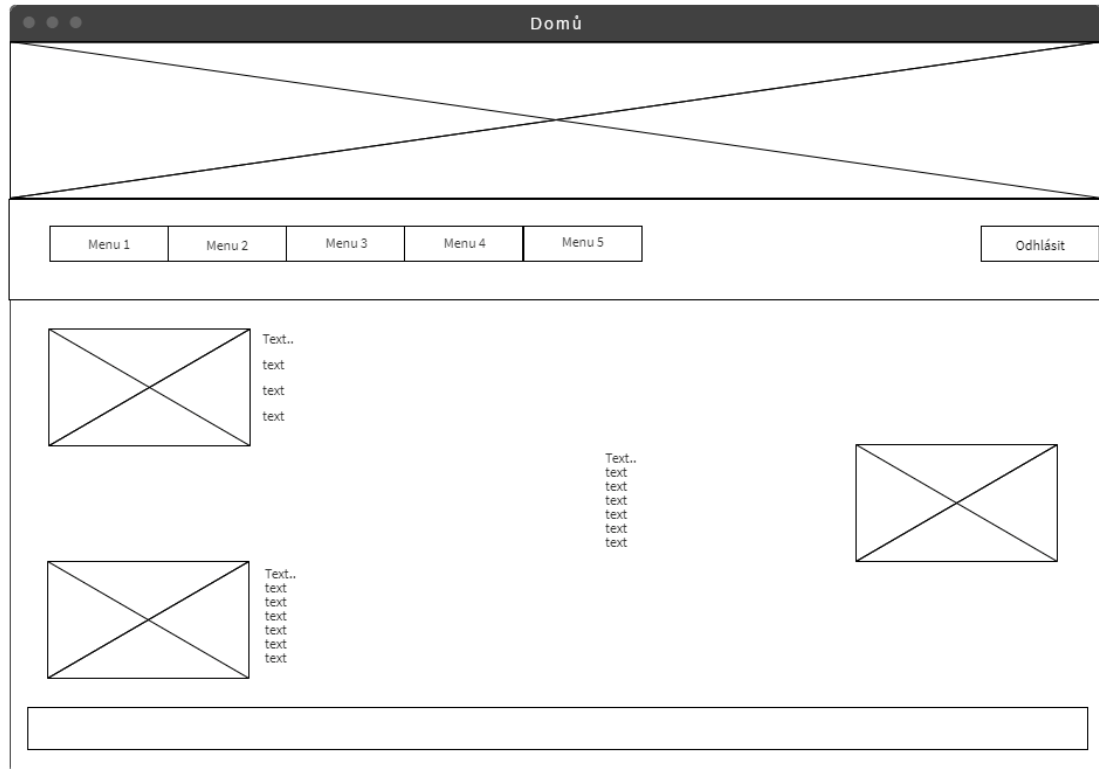
Logický design jsem navrhla a zpracovala prostřednictvím online webové aplikace <https://www.mockflow.com/>.

2.5.3.1 Domů – úvodní stránka

Záložka „Domů“ je úvodní stránkou, na která slouží pouze pro sdělení informací. Nyní obsahuje údaje o mé bakalářské práci, ale její pozdější záměr je poskytování informací ohledně útulků, různých článků o zvířatech a dalších tématech týkajících se zvířat.



Obrázek 2 - Wireframe úvodní stránky



Obrázek 3 - Wireframe úvodní stránky (po přihlášení)

2.5.3.2 K adopci

Stránka „K adopci“ slouží jako katalog všech inzerovaných zvířat k adopci a je přístupná i pro nepřihlášené uživatele. Uživatelé si zde mohou prohlédnout jednotlivé profily zvířat k adopci, přičemž každý profil obsahuje fotku zvířete, základní údaje a popis. Součástí popisu mohou být podrobnější informace o zvířeti, ale primárně kontakt na uživatele, který zvíře inzeruje. Pro přehlednější zobrazení a snadnější vyhledávání může uživatel použít filtr, který nabízí třídění podle druhu, pohlaví, věku a lokality zvířete. Lokalita je členěná podle Krajů ČR.



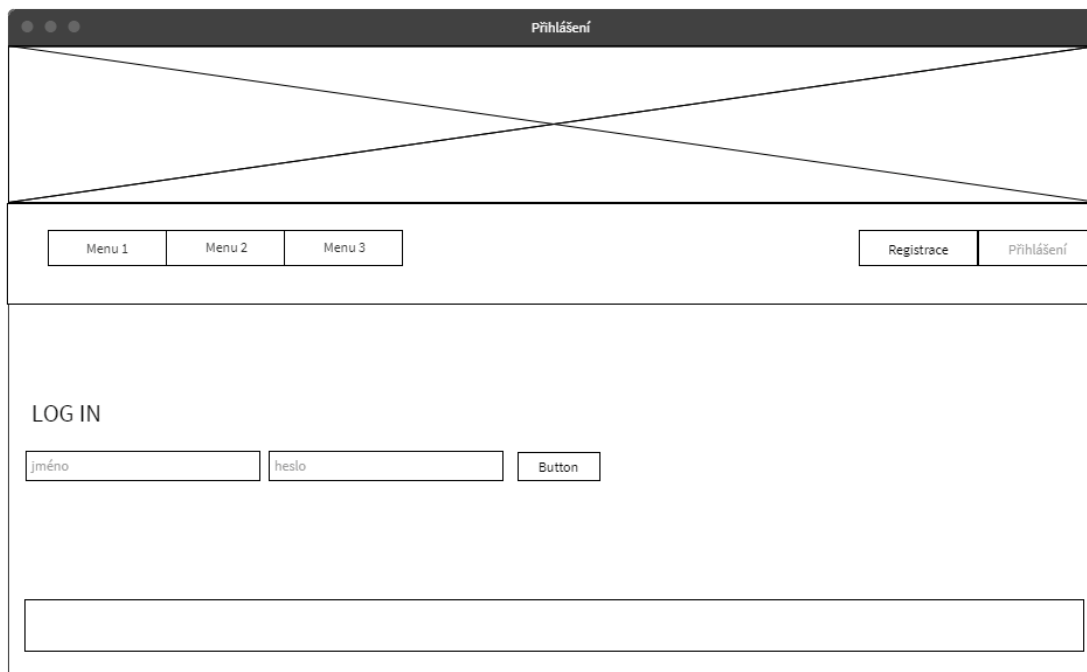
Obrázek 4 - Wireframe pro zobrazení zvířat k adopci



Obrázek 5 - Wireframe pro zobrazení zvířat k adopci (po přihlášení)

2.5.3.3 Přihlášení

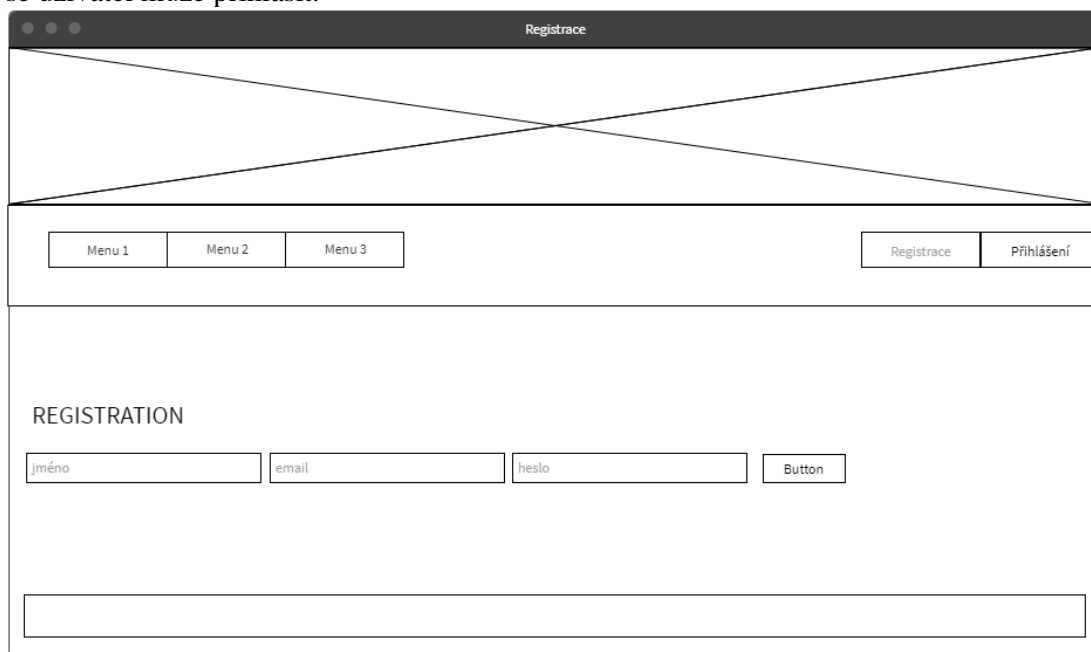
Pro úspěšné přihlášení musí uživatel vyplnit jméno a heslo, které si nastavil při registraci. Po přihlášení se panel nástrojů rozšíří a uživatel má přístup ke vkládání inzerátů. Přihlášení je tedy zapotřebí pouze pro uživatele, kteří chtějí inzerovat zvířata k adopci.



Obrázek 6 - Wireframe pro přihlášení do portálu

2.5.3.4 Registrace

Pokud má uživatel zájem vložit inzerát na tento portál, musí se zaregistrovat. Pro vytvoření nového účtu, je nutné vyplnit jméno, email a vytvořit heslo. Ihned po dokončení registrace se uživatel může přihlásit.



Obrázek 7 - Wireframe pro vytváření nového účtu

2.5.3.5 Moje inzeráty

Záložka „Moje inzeráty“ slouží jako evidence inzerovaných zvířat, která vložil přihlášený uživatel.

The wireframe shows a browser window titled "Moje inzeráty". The main content area is currently empty, indicated by a large 'X' across the top. Below this is a navigation bar with five menu items: "Menu 1", "Menu 2", "Menu 3", "Menu 4", and "Menu 5", followed by an "Odhlásit" button. The main form area contains a "Jméno" label above a placeholder image box. To the right of the image box is a "popis.." label. Below the image box are five labels: "Datum:", "Druh:", "Pohlaví:", "Stáří:", and "Kraj:". At the bottom of the form is a wide, empty text input field.

Obrázek 8 - Wireframe pro vložení inzerce přihlášeného uživatele

2.5.3.6 Vložit inzerát

Vložení inzerátu je možné pouze pro registrované a úspěšně přihlášené uživatele. Pro nahrání inzerátu zvířete k adopci je nutné nahrát fotku zvířete, vyplnit jeho jméno, vybrat druh zvířete z nabídky kategorií, dále zaškrtnout pohlaví, určit stáří a vybrat kraj, ze kterého zvíře pochází. Nedílnou součástí vložení inzerátu je také popis, ve kterém je vhodné napsat kontaktní údaje osoby, která zvíře inzeruje.

Vložit inzerát

Menu 1 Menu 2 Menu 3 Menu 4 Menu 5 Odhlásit

Nahraj obrázek: *

Jméno: *

Vyber zvíře: *

Pohlaví: *
 Male
 Female

Stáří zvířete: *

Kraj: *

popis..

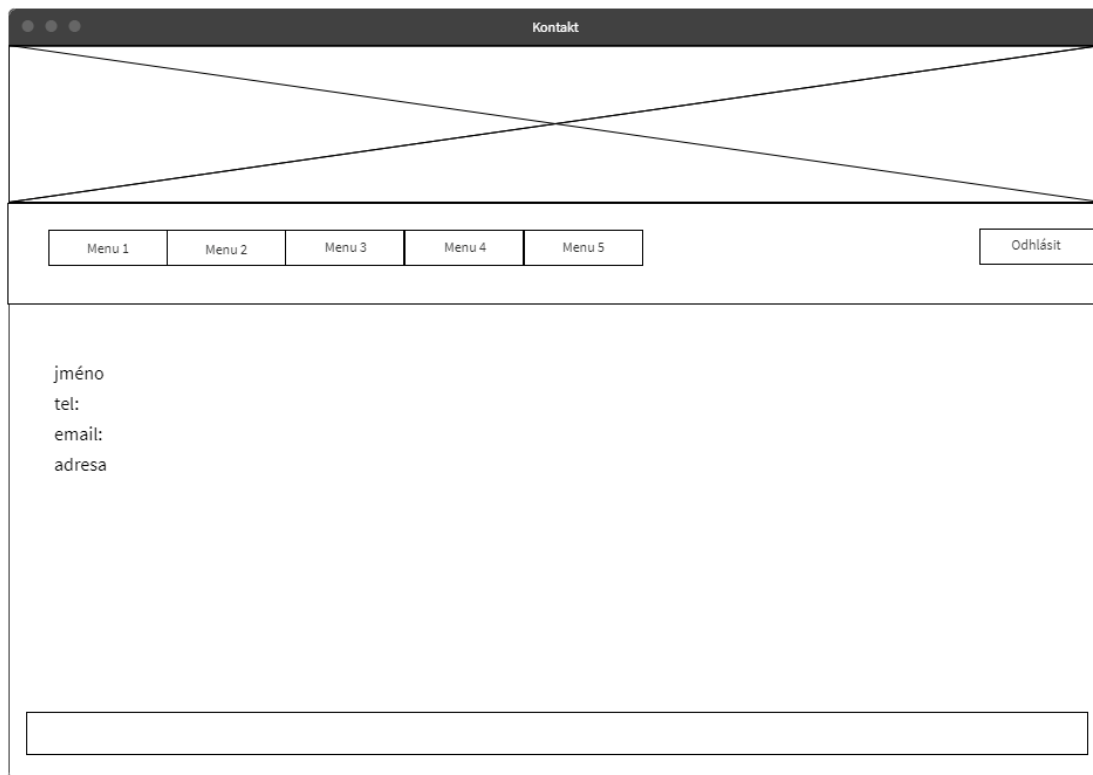
Obrázek 9 - Wireframe pro nahrání inzerátu

2.5.3.7 Kontakt

Na stránce „Kontakt“ jsou uvedeny kontaktní údaje na správce portálu, který zodpovídá na veškeré dotazy a zajišťuje kontrolu a editaci vložených inzerátů.



Obrázek 10 - Wireframe pro stránku s kontaktem



Obrázek 11 - Wireframe pro stránku s kontaktem (po přihlášení)

2.6 Metodika řešení

Při vývoji webové aplikace jsem používala již zmíněné nástroje. Aplikace je konstruována podle osvědčených metod a pomocí nástrojů vhodných pro vývoj webových aplikací.

2.6.1 Xampp - instalace

Nainstalovala jsem verzi Xampp 7.3.11 z oficiálních stránek apachefriends.org a následně jsem díky tomu nainstalovala apache distribuci obsahující php a mysql, kterou potřebuji k běhu mé webové aplikace. Pro správný běh webové aplikace je zapotřebí si nejdříve otevřít Xampp control panel, používám verzi 3.2.4, kde stisknutím „start“ spustím Apache a MySQL. (11)

2.6.2 Síťový port

Síťový port je číslo v rozmezí hodnot 0 až 65535 a označení se používá pro komunikaci mezi jednotlivými programy pomocí protokolů TCP a UDP. (12)

Tabulka 5 –Síťové porty (12)

Port	Protokol	Popis
21, 20	FTP, FTP- data	Přenos souborů (řídící a datové spojení)
22	SSH	Secure shell - šifrovaná obdoba protokolu telnet, přenosy souborů, forwardování portů
23	Telnet	Vzdálený textový terminál – nešifrovaná komunikace
25	SMTP	Simple Mail Transfer Protocol – přenos elektronické pošty
53	DNS	Domain Name System – překlad doménových jmen na IP adresy a zpět
80	HTTP	HyperText Transfer Protocol – přenos WWW stránek i jiných dat
110	POP3	Post Office Protocol version 3 – stahování elektronické pošty
143	IMAP	Internet Message Access Protocol 4 – vzdálená správa poštovní schránky s elektronickou poštou
443	HTTPS	Šifrovaný přenos HTTP protokolu přes TLS

2.6.3 Apache

Apache HTTP Server, hovorově nazývaný Apache je bezplatný a otevřený software pro více platform pro webové servery, vydaný za podmínek licence Apache License 2.0. Při spuštění Apache používám port 80. Hyper Text Transfer Protocol (HTTP) - port používaný pro webový provoz pro přenos WWW stránek.

2.6.4 MySQL

MySQL je databázový systém pro řízení relačních tabulek. V rámci mé práce využívám systém phpMyAdmin, který umožňuje jednoduchou a přehlednou správu databáze pomocí webového rozhraní. Při spuštění MySQL nastavuji port 3306. Port 3306 je výchozí port pro klasický protokol MySQL (port), který používá klient mysql, MySQL Connector a nástroje jako mysqldump a mysqlpump.

2.7 Implementace

2.7.1 Návrh databáze

2.7.1.1 ERD – Diagram Vztahu Entit

Nejoblíbenější technika vyjádření návrhu dat pro datově orientované systémy.

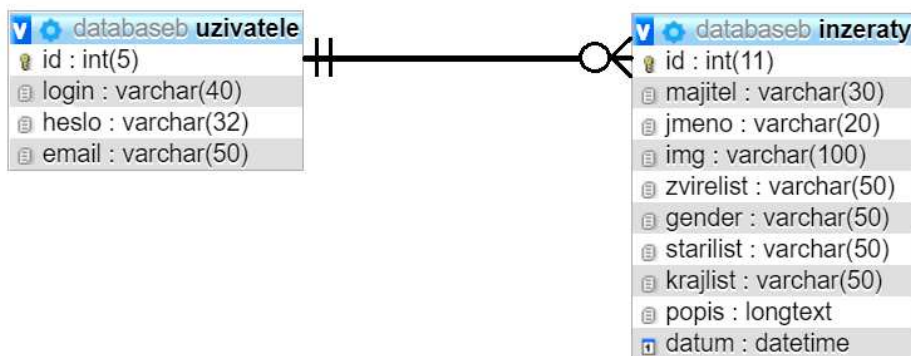
ERD zobrazuje typy objektů (entit), o kterých v systému ukládáme data, typy vztahů mezi těmito objekty a někdy typy atributů ukládaných o každém objektu.

Níže na obrázku 12 je vytvořený diagram vztahu entit „database“, který zobrazuje vztah mezi tabulkou „uzivatele“ a tabulkou „inzeraty“.

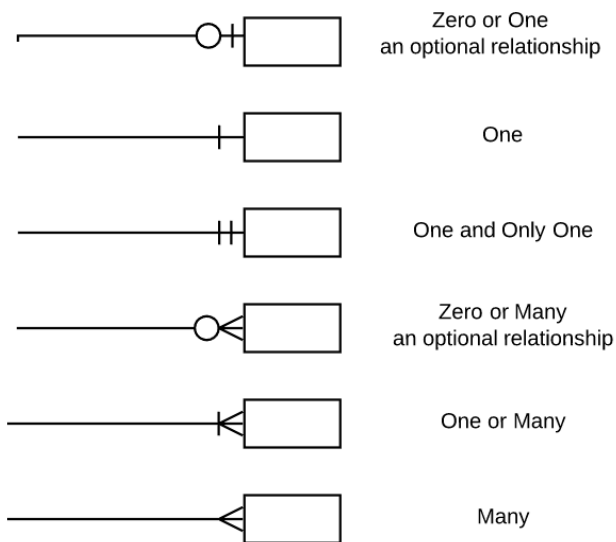
Vztah je graficky znázorněn pomocí symbolů, které znázorňují, o jaký vztah se jedná viz.

obrázek 13, kde jsou zobrazeny jednotlivé symboly s popisem vztahu, který daný symbol znázorňuje.

Vztah mezi entitami „uzivatele“ a „inzeraty“ jsem určeny jako **one and only one to zero or many**. Entita „uzivatele“ může být jedna, a pouze jedna pro nula nebo více entit „inzeraty“. Znamená to, že jeden konkrétní uživatel může mít vložených 0 nebo více inzerátů, a zároveň každý inzerát může být vložený pouze jedním konkrétním nezaměnitelným uživatelem.



Obrázek 12 - Diagram vztahu entit



Obrázek 13 – Symboly vztahů entit (13)

2.7.2 Vytvoření tabulky

Tabulky jsou tvořeny příkazem `CREATE TABLE` v jazyce SQL (Structured Query Language). V případě volby tohoto příkazu explicitně definujeme všechny komponenty tabulky, u kterých určíme datové typy, primární/cizí klíče atd.

2.7.2.1 Datové typy

Integer identity (zkratka „int“) je celočíselný datový typ, který definuje, že hodnota položky celočíselného typu je matematické celé číslo, kterému odpovídá.

Varchar označuje znakové řetězce až o 255 bajtech do polí proměnlivé délky. Data mohou být tvořena písmeny, čísly a symboly. Číslo u tohoto datového typu označuje maximální délku v bajtech.

Datový typ `longtext` se používá pro delší text, nerozlišuje velikost písmen a délka řetězce je max. 4 294 967 295 znaků.

Datetime označuje datum a čas. Rozsah od 1000-01-01 00:00:00 do 9999-12-31 23:59:59 (format = "RRRR-MM-DD HH:MM:SS")

Not null označuje, že hodnota nesmí být prázdná či-li nepůjde do ní vložit hodnota NULL.

2.7.2.2 Úložné databázové stroje (storage engine)

MySQL zahrnuje více typů databázových úložišť, které se liší možnostmi, použitím a způsobem ukládání dat do souborů.

Typ InnoDB používá transakce a je optimalizován pro relativně větší rychlost dotazů `INSERT` a `UPDATE` oproti typu MyISAM a pro práci s velkým objemem dat.

Typ tabulek MyISAM nepodporuje transakce, ale je velmi rychlý a optimalizován pro dotazy `SELECT`.

```
CREATE TABLE `inzeraty` (
```

```

`id` int(11) NOT NULL,
`majitel` varchar(30) NOT NULL,
`jmeno` varchar(20) NOT NULL,
`img` varchar(100) NOT NULL,
`zvirelist` varchar(50) NOT NULL,
`gender` varchar(50) NOT NULL,
`starilist` varchar(50) NOT NULL,
`krajlist` varchar(50) NOT NULL,
`popis` longtext NOT NULL,
`datum` datetime NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

```

Zdrojový kód 1 – Příkaz SQL pro vytvoření tabulky „inzeraty“

```

CREATE TABLE `uzivatele` (
  `id` int(5) NOT NULL,
  `login` varchar(40) NOT NULL,
  `heslo` varchar(32) NOT NULL,
  `email` varchar(50) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

```

Zdrojový kód 2 - Příkaz SQL pro vytvoření tabulky „uzivatele“

2.7.3 Připojení k databázi

Pro připojení k databázi je využitý ovladač relační databáze MySQLi, který je používán ve skriptovacím jazyce PHP k poskytování rozhraní s databázemi MySQL.

V ukázkovém zdrojovém kódu 3 je zobrazena funkce, obsahující parametr hostname, který určuje název hostitele (localhost), parametry uživatelského jména (root) a hesla (""), které určují pověření pro přístup k serveru MySQL a parametr databáze, kde je určena výchozí databáze pro použití při provádění dotazů.

Součástí tohoto zdrojového kódu je také kontrola připojení k databázi, kde se vrací chybový kód z posledního volání spojení.

```

<?php
$mysqli = new mysqli("localhost","root","","database");

// Kontroluje připojení
if ($mysqli -> connect_errno) {
  echo "Failed to connect to MySQL: " . $mysqli -> connect_error;
  exit();
}
?>

```

Zdrojový kód 3 - Připojení k databázi MySQL v PHP

2.7.4 Filtr zvířat k adopci

Webová stránka načítá informace z databáze pomocí AJAX.

Zdrojový kód 4 (adopce) představuje funkci s názvem "showUser ()".

Funkce je provedena, když uživatel vybere osobu v rozevíracím seznamu a je spuštěna událostí onchange.

Metoda Document getElementById () vrací objekt Element představující prvek, jehož vlastnost id odpovídá zadanému řetězci.

Kód funguje tak, že se nejprve zkontroluje, zda je vybrána osoba.

Pokud není vybrána žádná osoba (str == ""), vymaže se obsah txtHint a ukončí se funkce.

Pokud je vybrána osoba, postupuje se tak, že se vytvoří objekt XMLHttpRequest a funkce, která se má provést, když je odpověď serveru připravena a odešle se požadavek do souboru na serveru.

Stránka na serveru, která je volaná JavaScriptem je soubor PHP s názvem „najdiZvire.php“.

Zdrojový kód v souboru „najdiZvire.php“ spustí dotaz na databázi MySQL a vrátí výsledek v tabulce HTML.

Vysvětlení tedy zní, že když je dotaz odeslán z JavaScriptu do souboru PHP, tak PHP otevírá připojení k serveru MySQL, následně je nalezena správná osoba a vytvoří se tabulka HTML, která se vyplní daty a odešle se zpět na zástupný symbol „txtHint“.

Ukázkový zdrojový kód 5 (najdi zvíře) obsahuje příkaz switch, pomocí kterého se vybere jeden z mnoha bloků kódu, které mají být provedeny.

Nejprve je jediný výraz n (nejčastěji proměnnou), který je vyhodnocen jednou. Hodnota výrazu se poté porovná s hodnotami pro každý případ ve struktuře. Pokud existuje shoda, provede se blok kódu přidruženého k danému případu.

Pro zabránění automatickému spuštění kódu do dalšího případu se používá break.

Výchozí příkaz se použije, pokud není nalezena shoda.


```

<?php include 'header.php'; ?>
<?php require('conn.php'); ?>
<script>
function showUser(str) {

    if (str == "") {
        document.getElementById("txtHint").innerHTML = "";
        return;
    } else {
        if (window.XMLHttpRequest) {
            // kód pro IE7+, Firefox, Chrome, Opera, Safari
            xmlhttp = new XMLHttpRequest();
        } else {
            // kód pro IE6, IE5
            xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
        }
        xmlhttp.onreadystatechange = function() {
            if (this.readyState == 4 && this.status == 200) {
                document.getElementById("txtHint").innerHTML = this.responseText;
            }
        };

        xmlhttp.open("GET","najdiZvire.php?q="+druh.value+"&w="+gender.value+"&e="+stari.
value+"&r="+kraj.value);
        xmlhttp.send();
    }
}
</script>
</head>
<body>

<form name="search">
<select name="users" id="druh" onchange="return showUser(9)">

    <option value="0">Vyberte druh</option>
    <option value="1">Kočka</option>
    <option value="2">Kůň</option>
    <option value="3">Pták</option>
    <option value="4">Pes</option>
    <option value="5">Malý savec</option>
    <option value="6">Plaz</option>
    <option value="7">Hmyz</option>
</select>
<select name="users" ID="gender" onchange="return showUser(9)" >
<option value="0">Vyberte pohlaví</option>
<option value="88">samec</option>
<option value="99">samička</option>

```

```

</select>
<select id="stari" name="users" onchange="return showUser(9)">
<option value="0">-</option>
<option value="111">do 1 roku</option>
<option value="222">1-3 roky</option>
<option value="333">3-5 let</option>
<option value="444">5 a více</option>
</select>
<select id="kraj" name="users" onchange="return showUser(9)">
<option value="0">-</option>
<option value="1111">Středočeský kraj</option>
<option value="2222">Jihočeský kraj</option>
<option value="3333">Západočeský kraj</option>
<option value="4444">Severočeský kraj </option>
<option value="5555">Východočeský kraj </option>
<option value="6666">Jihomoravský kraj </option>
<option value="7777">Severomoravský kraj </option>
<option value="8888">Praha</option>

</select>

</form>
<br>
<div id="txtHint"><b>Zadejte parametry vyhledávání...</b></div>
<?php include 'footer.php'; ?>

```

Zdrojový kód 4 – Adopce

```

<style>
table {
width: 100%;
border-collapse: collapse;
}

table, td, th {
border: 1px solid black;
padding: 5px;
}

th {text-align: left;}
</style>

<?php require('conn.php'); ?>
<?php
$q = intval($_GET['q']);
switch ($q ) {
case 1:
$q ="Kočka";

```

```

    break;
case 2:
    $q = "Kůň";
    break;
    case 3:
    $q = "Pták";
    break;
    case 4:
    $q = "Pes";
    break;
    case 5:
    $q = "Malý savec";
    break;
    case 6:
    $q = "Plaz";
    break;
    case 7:
    $q = "Hmyz";
    break;
default:
    $q = "%";
}

$w = intval($_GET['w']);

switch ($w ) {
    case 88:
        $w = "male";
        break;
    case 99:
        $w = "female";
        break;
    default:
        $w = "%";
}

$e = intval($_GET['e']);

switch ($e ) {
    case 111:
        $e = "1";
        break;
    case 222:
        $e = "2";
        break;
    case 333:
        $e = "3";
        break;
}

```

```

        case 444:
            $e = "4";
            break;
        default:
            $e = "%";;
    }

    $r = $_GET['r'];
    switch ($r ) {
        case 1111:
            $r = "Středočeský kraj";
            break;
        case 2222:
            $r = "Jihočeský kraj";
            break;
            case 3333:
                $r = "Západočeský kraj";
                break;
            case 4444:
                $r = "Severočeský kraj";
                break;
            case 5555:
                $r = "Východočeský kraj";
                break;
            case 6666:
                $r = "Jihomoravský kraj";
                break;
            case 7777:
                $r = "Severomoravský kraj";
                break;
            case 8888:
                $r = "Praha";
                break;
        default:
            $r = "%";;
    }

    $sql="SELECT * FROM `inzeraty`
WHERE
(zvirelist LIKE '$q') AND
(gender LIKE '$w') AND
(starilist LIKE '$e') AND
(krajlist LIKE '$r')
";

$result = mysqli_query($mysqli,$sql);

while($row = mysqli_fetch_array($result)) {

```

```

        echo '<div class="zvirata">';
echo "<h3>" . $row['jmeno'] . "</h3><div class='oneline'>";

echo '<div style="float:left;"></div>';
echo '        <div style="float:right; margin-left: 30px;"><p>' . $row['popis'] .
'</p></div></div>';
echo '        <div><div class="oneline"><b>Datum : </b>' . $row['datum'] . '</div>';
echo '        <div class="oneline"><b>Druh : </b>' . $row['zvirelist'] . '</div>';
echo '        <div class="oneline"><b>Pohlaví : </b>' . $row['gender'] . '</div>';
echo '        <div class="oneline"><b>Stáří : </b>' . $row['starilist'] . '</div>';
echo '        <div class="oneline"><b>Kraj : </b>' . $row['krajlist'] . '</div></div>';

echo '</div>';

        echo "</div>";
}

mysqli_close($mysqli);
?>
</body>
</html>

```

Zdrojový kód 5 - Najdi zvíře

2.7.5 Ukázkový kód pro vložení inzerátu

V tomto ukázkovém zdrojovém kódu je pro vložení inzerátu nejprve vyžadováno připojení k databázi. Pomocí „require“ se vloží kód PHP s připojením k databázi ze souboru „conn.php“.

PHP \$_POST je super globální proměnná PHP, která se používá ke shromažďování dat formulářů po odeslání formuláře HTML s metodou = "post". \$_POST je také široce používán k předávání proměnných.

PHP \$_REQUEST je super globální proměnná PHP, která se používá ke sběru dat po odeslání formuláře HTML.

```

<?php include 'header.php'; ?>
<?php

if(!isset($_SESSION["username"])){

header("Location: login.php");
exit(); }
?>
<?php require('conn.php'); ?>

<?php

```

```

// definuje proměnné a nastavuje na prázdné hodnoty
$uploadErr = $nameErr = $zvirelistErr = $genderErr = $starilistErr = $krajlistErr =
$popisErr = "";
$upload = $name = $zvirelist = $gender = $starilist = $krajlist = $popis = "";
$ERROR="";
$majitel="";
$target_file = "";
$target_dir = "uploads/";
$img="";
if(isset($_POST["submit"])||!empty($_FILES["fileToUpload"]["name"])) {
    $target_file = $target_dir.basename($_FILES["fileToUpload"]["name"]);
    $imageFileType = pathinfo($target_file,PATHINFO_EXTENSION);
    $check = getimagesize($_FILES["fileToUpload"]["tmp_name"]);
    if($check !== false) {
        $check["mime"];
        /* echo "Soubor je obrázek - " . . . .";*/
        $uploadOk = 1;
        // Kontrolujte, zda soubor již existuje
    }
    // Kontrolujte, zda soubor již existuje
    if (file_exists($target_file)) {
        $uploadErr = "Omlouváme se, tento obrázek již máme v databázi.";
        $uploadOk = 0;
    }
    // Kontroluje velikost souboru
    if ($_FILES["fileToUpload"]["size"] > 500000) {
        $uploadErr = "Omlouváme se váš soubor je příliš veliký.";
        $uploadOk = 0;
    }
    // Povoluje určité formáty souborů
    if($imageFileType != "jpg" && $imageFileType != "png" && $imageFileType != "jpeg"
    && $imageFileType != "gif" ) {
        $uploadErr = "Omlouváme se, musíte nahrávat pouze obrázky ve formátu JPG, JPEG,
    PNG & GIF.";
        $uploadOk = 0;
    }

    if ($uploadOk == 1) {
        $ERROR++;
    }

    if (empty($_POST["jmeno"])) {
        $nameErr = "Nutný název";
    }
    else {
        $name = test_input($_POST["jmeno"]);
        $ERROR++;
    }
}

```

```

if (stripslashes($_POST["zvirelist"])=="-") {
    $zvirelistErr = "Vyber zvíře";
} else {
    $zvirelist = stripslashes($_REQUEST['zvirelist']);
    $ERROR++;
}

if (empty($_POST["gender"])) {
    $genderErr = "Zvol pohlaví";
} else {
    $gender = test_input($_POST["gender"]);
    $ERROR++;
}

if (stripslashes($_POST["starilist"])=="-") {
    $starilistErr = "Zvol stáří";
} else {
    $starilist = test_input($_POST["starilist"]);
    $ERROR++;
}

if (stripslashes($_POST["krajlist"])=="-") {
    $krajlistErr = "Zvol kraj";
} else {
    $krajlist = test_input($_POST["krajlist"]);
    $ERROR++;
}

if (stripslashes($_POST["popis"])=="Popis...") {
    $popisErr = "Nutný popis";
} else {
    $popis = test_input($_POST["popis"]);
    $ERROR++;
}
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}

```

```

if($ERROR==7){
    if ($uploadOk == 0) {
        // $uploadErr = "Litujeme, váš soubor nebyl nahrán.";
        // Pokud je vše v pořádku, zkuste nahrát soubor
    } else {
        if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $target_file)) {
            $uploadErr = "The file ". basename( $_FILES["fileToUpload"]["name"]). " byl
nahrán.";
                $ERROR++;
            } else {
                $uploadErr = "Omlouváme se, vyskytla se chyba s nahráváním souboru.";
            }
        }
    }

    $user = $_SESSION["username"];

    $result = mysqli_query($mysqli,"SELECT `id` FROM `uzivatele` WHERE login =
''.$user.'''");
if (mysqli_num_rows($result) > 0) {
    // výstupní data každého řádku
    while($row = mysqli_fetch_assoc($result)) {
        $majitel= $row["id"];
    }
} else {
    echo "0 results";
}

$img = basename( $_FILES["fileToUpload"]["name"] ) ;

    $name = mysqli_real_escape_string($mysqli,$name);
    $zvirelist = mysqli_real_escape_string($mysqli,$zvirelist);
    $gender = mysqli_real_escape_string($mysqli,$gender);
    $starilist = mysqli_real_escape_string($mysqli,$starilist);
    $krajlist = mysqli_real_escape_string($mysqli,$krajlist);
    $popis = mysqli_real_escape_string($mysqli,$popis);
    $query = "INSERT into `inzeraty` (majitel ,jmeno ,img ,zvirelist, gender, starilist,
krajlist,popis)
VALUES ('$majitel', '$name','$img','$zvirelist', '$gender', '$starilist', '$krajlist', '$popis')";
    $result = mysqli_query($mysqli,$query);
    if($result){
        echo '<span class="error">Inzerát vložen</span>';
    }else{
        echo "Chyba?";
    }
}

?>
<style>
.error {color: #FF0000;}

```



```

</style>
<div class="form">
<h1>Přidání zvířete</h1>
<p><span class="error">* Nutno vyplnit</span></p>
<form          method="post"          action="<?php          echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>" enctype="multipart/form-data">
<label for="fileToUpload">Nahraj Obrázek</label><span class="error">* <?php echo
$uploadErr;?></span><br>
<input type="file" name="fileToUpload" ><br>
<label for="jmeno">Jméno:</label><br>
<input type="text" name="jmeno">
  <span class="error">* <?php echo $nameErr;?></span>
  <br><br>
<label for="zvirata">Vyber zvíře:</label>
<select id="zvirata" name="zvirelist">

  <option value="-">-</option>
  <option value="Kočka">Kočka</option>
  <option value="Kůň">Kůň</option>
  <option value="Pták">Pták</option>
  <option value="Pes">Pes</option>
  <option value="Malý savec">Malý savec</option>
  <option value="Plaz">Plaz</option>
  <option value="Hmyz">Hmyz</option>
</select><span class="error">* <?php echo $zvirelistErr;?></span><br>

<label for="pohlavi">Pohlaví:</label><span class="error">* <?php echo
$genderErr;?></span><br>
<input type="radio" id="male" name="gender" value="male">
  <label for="male">samec</label><br>
  <input type="radio" id="female" name="gender" value="female">
  <label for="female">amička</label><br>
<label for="stari">Stáří zvířete:</label>
<select id="stari" name="starilist">
  <option value="-">-</option>
  <option value="1">do 1 roku</option>
  <option value="2">1-3 roky</option>
  <option value="3">3-5 let</option>
  <option value="4">5 a více</option>
</select><span class="error">* <?php echo $starilistErr;?></span><br>

<label for="kraj">Kraj:</label>
<select id="kraj" name="krajlist">
  <option value="-">-</option>
  <option value="Středočeský kraj">Středočeský kraj</option>
  <option value="Jihočeský kraj">Jihočeský kraj</option>
  <option value="Západočeský kraj">Západočeský kraj</option>
  <option value="Severočeský kraj">Severočeský kraj </option>
  <option value="Východočeský kraj">Východočeský kraj </option>

```

```
<option value="Jihomoravský kraj">Jihomoravský kraj </option>
<option value="Severomoravský kraj">Severomoravský kraj </option>
<option value="Praha">Praha</option>

</select><span class="error">* <?php echo $krajlistErr;?></span><br>

<textarea rows="4" cols="50" name="popis" >
Popis by měl obsahovat Vaše kontaktní údaje...</textarea> <span class="error">* <?php
echo $popisErr;?></span>

<input type="submit" name="submit" value="submit" />
</form>
</div>

<?php include 'footer.php'; ?>
```

Zdrojový kód 6 - Vložení inzerátu

Závěr

Hlavním tématem této bakalářské práce bylo navrhnutí a realizace řešení evidence opuštěných a ztracených zvířat. Řešením bylo vytvoření portálu zvířat, který je možné využít jak pro zobrazení zvířat k adopci a jejich následné adopce, tak pro nahrání inzerátů. Vložit inzerát je umožněno pouze registrovaným a úspěšně přihlášeným uživatelům.

Využití portálu je také míněno pro zvířecí útulky, cílem totiž bylo vytvořit centralizovanou databázi opuštěných zvířat po celé ČR a využít jí může kdokoliv.

Teoretická část této práce se týkala popisu pojmů nezbytných nástrojů k vyhotovení webové aplikace.

Součástí praktické části byl návrh logického designu aplikace a její následná implementace, zahrnující bloky zdrojového kódu důležitých funkcí.

Ačkoliv jsem zadání splnila, tato webová aplikace by mohla být v budoucnu rozšířena o další funkce. Zejména se jedná o umožnění přihlášeným uživatelům editovat nahrané inzeráty a vytvořený účet po registraci.


3 Seznam použitých zdrojů

1. <https://www.svscr.cz/zdravi-zvirat/povinna-registrace-utulku-pro-zvirata/>
2. Shklar, L., & Rosen, R. (2009). Web Application Architecture: Principles, Protocols and Practices (2nd). Hoboken: John Wiley
3. KOSEK, Jiří. Webový server. HTML guru [online]. [cit. 2018-02-8]. Dostupné z: <http://htmlguru.cz/vystaveni-webovy-server.html>.
4. Základy relačních databází, jejich využití v programování webu. Grafická a multimediální laboratoř VŠE [online]. [cit. 2018-02-8]. Dostupné z: <http://gml.vse.cz/data/oppa-webdesign/zaklady-db.html>
5. <https://cs.wikipedia.org/wiki/XAMPP>
6. CONTRIBUTORS, phpMyAdmin. Translations. phpMyAdmin [online]. [cit. 2018-11-15]
7. <https://cs.wikipedia.org/wiki/Notepad%2B%2B>
8. <http://www.proskoly.cz/co-je-to-wireframe-webu/>
9. BRUCKNER, Tomáš. Tvorba informačních systémů: principy, metodiky, architektury. Praha: Grada, 2012. Management v informační společnosti. ISBN 978-80-247-4153-6.
10. <https://managementmania.com/cs/vodopadovy-model-waterfall-model>
11. NIXON, Robin. Learning PHP, MySQL & JavaScript: with jQuery, CSS & HTML5. 4th. Beijing: O'Reilly Media, 2014
12. <https://porty2.webgarden.cz/rubriky/porty/sitovy-port>
13. https://res.cloudinary.com/practicaldev/image/fetch/s--aMHSzLpu--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https://thepracticaldev.s3.amazonaws.com/i/gjqz00lrbmxqjx7i1wp.PNG

4 Přílohy

- Příloha A Vytvořená aplikace – úvodní strana
- Příloha B Vytvořená aplikace – filtr zobrazení zvířat k adopci
- Příloha C Vytvořená aplikace – vložení inzerátu
- Příloha D Soubor ve formátu ZIP se zdrojovými kódy


4.1 Příloha A Vytvořená aplikace – úvodní strana



[Domů](#) [K adopci](#) [Kontakt](#) [Registrace](#) [Přihlášení](#)

Vítejte na webové stránce!

Úvod




V souvislosti s mým zájmem o zvířata se mé téma týká jejich pomoci. Práce se dělí na dvě části, teoretickou a praktickou. V teoretické části jsou stručně popsány technologie, nezbytné pro vývoj webové aplikace.


Cíl práce

Cílem této bakalářské práce je navrhnout a implementovat webovou aplikaci pro portál zvířat, který bude umožňovat registrovaným uživatelům adoptovat či inzerovat zvířata hledající domov.

Metodika

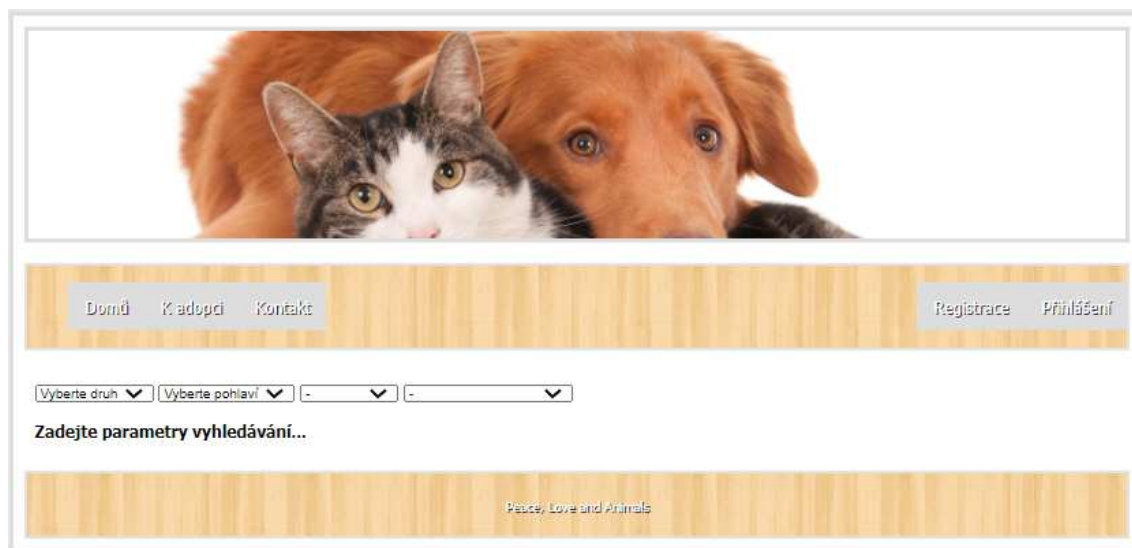


V rámci mé práce budu navrhovat a implementovat webovou aplikaci pro portál zvířat. Pro vytvoření aplikace využiji technologie pro vývoj webových aplikací a osvědčené postupy softwarového návrhu, které budu více zmiňovat v teoretické části. Praktická část bude především obsahovat ukázkové bloky zdrojového kódu a znázorněnou datovou strukturu.



Peace, Love and Animals

4.2 Příloha B Vytvořená aplikace – filtr zobrazení zvířat k adopci



Domů K adopci Kontakt Registrace Přihlášení

Vyberte druh Vyberte pohlaví - -

Zadejte parametry vyhledávání...

Peace, Love and Animals

4.3 Příloha C Vytvořená aplikace – vložení inzerátu



Domů K adopci Moje inzeráty Vložit inzerát Kontakt Odmítlát

Přidání zvířete

* Nutno vyplnit

Nahraj Obrázek
Choose File | No file chosen

Jméno: *

Vyber zvíře: - *

Pohlaví: *

Male
 Female

Stáří zvířete: - *

Kraj: - *

Popis... *

submit

Peace, Love and Animals

4.4 Příloha D Soubor ve formátu ZIP se zdrojovými kódy (vložen do UIS jako příloha k bakalářské práci)