



**FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE**

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Název:	Informační systém pro tábornický klub Oregon Kolín
Student:	Miroslav Mráček
Vedoucí:	Ing. Pavel Náplava, Ph.D.
Studijní program:	Informatika
Studijní obor:	Informační systémy a management
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2019/20

Pokyny pro vypracování

Tábornický klub Oregon Kolín je nezisková organizace, pořádající tábory, výpravy, schůzky a další aktivity nejen pro děti. Cílem práce je navrhnout a implementovat řešení, které usnadní a zefektivní vybrané procesy v této organizaci.

Postup práce je následující:

1. Proveďte analýzu fungování organizace a sběr požadavků pro návrh nového řešení.
2. Zpracujte rešerši existujících řešení a vyhodnocení vhodnosti použití některého z nalezených řešení ve srovnání s vlastním vývojem informačního systému na míru.
3. Navrhněte a implementujte řešení (dle předchozího výběru již existujícího nebo vyvinutého na míru), které pokryje minimálně klíčové požadavky organizace.
4. Ověřte funkčnost a uživatelské rozhraní formou uživatelských testů.
5. Vyhodnoťte náklady a přínosy implementovaného řešení, porovnejte stav před a po nasazení řešení.

Seznam odborné literatury

Dodá vedoucí práce.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
děkan

V Praze dne 29. ledna 2019



**FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE**

Bakalářská práce

Informační systém pro tábornický klub Oregon Kolín

Miroslav Mráček

Katedra softwarového inženýrství
Vedoucí práce: Ing. Pavel Náplava, Ph.D.

16. května 2019

Poděkování

Děkuji svému vedoucímu práce panu Ing. Pavlu Náplavovi za věcné a podnětné připomínky při vedení mé práce. Děkuji také členům tábornického klubu Oregon Kolín, za ochotu a spolupráci při uživatelském testování.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mé práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, avšak pouze k nevýdělečným účelům. Toto oprávnění je časově, teritoriálně i množstevně neomezené.

V Praze dne 16. května 2019

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2019 Miroslav Mráček. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.

Odkaz na tuto práci

Mráček, Miroslav. *Informační systém pro tábornický klub Oregon Kolín*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2019.

Abstrakt

Cílem práce je analyzovat současný stav, navrhnout a implementovat řešení, které usnadní a zefektivní vybrané procesy v tábornickém klubu Oregon Kolín. Jako řešení byl zvolen vývoj nového informačního systému s využitím PHP frameworku Nette. Data jsou ukládána do MySQL databáze. Pro objektovou reprezentaci dat z databáze je využíváno objektově relační mapování. Systém byl úspěšně nasazen a uživatelsky otestován. Nový systém podporuje pět ze šesti klíčových procesů. Práce je zakončena vyhodnocením přínosů a nákladů.

Klíčová slova informační systém, T.K. Oregon, web, Nette framework, PHP, Model-view-presenter, Objektově relační mapování, redakční systém

Abstract

The main goal of the bachelor thesis is to analyze contemporary state, design and implement solution, that will simplify and rise efficiency of selected processes in camping club Oregon Kolín. As a solution was selected a development of a new information system using PHP framework Nette. Data are saved to MySQL database. Data of database are in application represented

as objects with using of object-relational-mapping. The system was successfully deployed and tested by users. The new system supports five of six main processes. The bachelor thesis is ended by evaluation of benefits and costs.

Keywords information system, T.K. Oregon, web, Nette framework, PHP, Model-view-presenter, Object-relational mapping, Content management system

Obsah

Úvod	1
1 Cíl práce	3
2 Analýza organizace	5
2.1 Představení organizace	5
2.2 Analýza současných procesů	6
2.2.1 Přijetí nového člena	6
2.2.2 Placení členských příspěvků	6
2.2.3 Uskutečnění jednorázové akce	6
2.2.4 Uskutečnění pravidelné schůzky	7
2.2.5 Rozesílání hromadných emailů	7
2.2.6 Sdílení fotografií z táborů a výprav	8
2.3 Požadavky na nový systém	8
2.3.1 Funkční požadavky	8
2.3.1.1 F1 - Agenda akcí(výpravy, tábory a jiné akce)	8
2.3.1.2 F2 - Evidence členů	9
2.3.1.3 F3 - Evidovat placení členských příspěvků . .	10
2.3.1.4 F4 - Hromadné rozesílání emailů	10
2.3.1.5 F5 - Veřejná sekce webu	10
2.3.1.6 F6 - Foto galerie	10
2.3.1.7 F7 - Agenda schůzek	11
2.3.2 Nefunkční požadavky	11
2.3.2.1 N1 - Responzivní design	11
2.3.2.2 N2 - Snadná použitelnost	11
3 Rešerše existujících řešení	13
3.1 Hotový informační systém	13
3.2 Samostatné nástroje	14

3.3	Redakční systémy	14
3.4	Vyhodnocení řešerše	15
4	Návrh a realizace systému	17
4.1	Metodika vývoje	17
4.2	Případy užití	17
4.3	Použité technologie	19
4.3.1	PHP	19
4.3.2	HTML a CSS	19
4.3.3	Nette framework	19
4.3.4	Bootstrap 4	20
4.3.5	MySQL	20
4.3.6	Doctrine	20
4.4	Architektura systému	20
4.4.1	Model-View-Controller	20
4.4.1.1	Model	20
4.4.1.2	View	21
4.4.1.3	Controller	21
4.5	Databázová struktura	21
4.6	Grafické uživatelské rozhraní	22
4.7	Zefektivněné procesy	24
4.7.1	Přijetí nového člena	24
4.7.2	Placení členských příspěvků	24
4.7.3	Uskutečnění jednorázové akce	25
4.7.4	Rozesílání hromadných emailů	25
4.7.5	Sdílení fotografií z táborů a výprav	25
4.8	Testování vývojářem	26
4.9	Nasazení	26
5	Uživatelské testování	29
5.1	Výsledky testování	29
6	Vyhodnocení přínosů	31
6.1	Náklady na realizaci a provoz	31
6.2	Přínosy	31
	Závěr	33
	Bibliografie	35
	A Seznam použitých zkratk	37
	B Schéma databáze	39
	C Obsah příloženého CD	41

Seznam obrázků

2.1	Současná tabulka pro přihlašování na akce	7
4.1	Diagram zachycující případy užití a jejich aktéry	18
4.2	Ukázka GUI - menu ve veřejné sekci webu	22
4.3	Ukázka GUI - menu po přihlášení	23
4.4	Ukázka GUI - formulář pro zadávání platby příspěvků	23
4.5	Ukázka GUI - přihlašovací stránka na mobilním zařízení	24
4.6	Screenshoty používání ladícího nástroje Tracy	26
5.1	Test script - Vložení nové platby příspěvku	30
5.2	Test script - Vložení nové platby příspěvku	30

Seznam tabulek

6.1	Rozpis časové náročnosti jednotlivých činností	31
-----	--	----

Výpis kódu

1	Ukázka rozhraní fasády v modelové vrstvě	21
2	Ukázka latte šablony ve vrstvě view	22

Úvod

Při volbě tématu mé bakalářské práce jsem se zaměřoval především na to, aby má práce byla praktická a řešila reálné problémy.

Již od svého dětství jsem členem tábornického klubu Oregon Kolín. Dříve v roli dítěte a posledních téměř 10 let i v roli vedoucího. Jako vedoucí jsem měl možnost pozorovat, s jakými problémy se organizace potýká. Problémy souvisejí s rozrůstáním organizace, ale také se změnou doby a legislativními opatřeními, která se vztahují i na tuto dobročinnou neziskovou organizaci. Nová doba však kromě nových problémů přinesla i nové technologie a to především masivní rozšíření chytrých mobilních telefonů a internetového připojení do naprosté většiny českých rodin.

Rozhodl jsem se tedy podrobněji analyzovat situaci a realizovat řešení, které by s pomocí vhodných technologií vyřešilo některé hlavní problémy a zefektivnilo vybrané procesy v této organizaci.

Cíl práce

Cílem této práce je navrhnout a implementovat řešení, které usnadní a zefektivní vybrané procesy v tábornickém klubu Oregon Kolín. Pro splnění toho cíle je třeba provést následující kroky, které odpovídají i struktuře této práce.

Nejprve provedu analýzu současných procesů a ve spolupráci s hlavními zástupci organizace specifikujeme požadavky, které by mělo nové řešení splňovat. Dále zpracuji řešerši, ve které prozkoumám existující řešení, která by mohla stanovené požadavky splňovat. Dle výsledků řešerše rozhodnu, zda je vhodnější použít některé z nalezených řešení, nebo vytvořit nový informační systém na míru podle stanovených požadavků. Zvolené řešení musí pokrývat minimálně klíčové požadavky organizace. Řešení implementuji, nasadím a jeho funkčnost a použitelnost otestuji na reálných uživateli. Na závěr provedu vyhodnocení přínosů a nákladů implementovaného řešení a porovnáím stav před a po nasazení řešení.

Analýza organizace

V následujících kapitolách nejdřív představím organizaci a problémy se kterými se potýká, analyzuji současné procesy a představím dohodnuté požadavky na nové řešení. Nakonec provedu rešerši existujících řešení na základě které rozhodnu, zda je vhodné využít některé z nalezených řešení, nebo navrhnout nové řešení na míru.

2.1 Představení organizace

Táborský klub Oregon Kolín je nezisková organizace zaměřující se na práci s dětmi. Vede je k samostatnosti, sportu, hrám, přírodě, moudrostem indiánů a dovednostem táborníků. Děti i vedoucí se scházejí každé léto na tábore, v průběhu roku na zhruba desítky výprav a jednou týdně na schůzce v klubovně. Oddíl existuje už od roku 1989 a v roce 1990 vstoupil do České táborské unie, jejíž součástí je dodnes. [1]

Oregon se dělí aktuálně na 3 družiny. V každé družině je zhruba 15 dětí podobného věku. Celkem má oddíl 50 dětí a zhruba 20 vedoucích. Zhruba polovina vedoucích je aktivních a vedou pravidelně jednotlivé družiny (družinový vedoucí) a druhá polovina se účastní jen tábora a některých výprav.

Chod organizace aktuálně komplikují hlavně tyto nedostatky:

- nejednotný systém evidování důležitých údajů o členech;
- nejednotný systém pro placení a evidování zaplacených příspěvků;
- komplikovaná příprava podkladů k výpravám;
- absence přehledného harmonogramu nadcházejících akcí;
- přidávání fotografií na web je závislé na jediném člověku
- a problémy se sdílením seznamu emailových adres pro hromadné rozesílání emailů.

2.2 Analýza současných procesů

V předchozí kapitole jsem zmínil seznam hlavních problémů, se kterými se organizace potýká a zde provedu procesní analýzu, která tyto problémy detailněji objasní v kontextu klíčových procesů v organizaci.

2.2.1 Přijetí nového člena

Proces přijetí nového člena začíná tím, že uchazeč nebo jeho rodiče kontaktují některého z družinových vedoucích, aby zjistili podrobnější informace o oddílu a ověřili si, zda je možné nového člena přijmout. Nový člen je přijat, pokud to aktuální kapacita družiny dovoluje a přijetí je schváleno vedoucími dané družiny. Od člena a jeho rodičů je vyžadováno vyplnění přihlášky ve které se uvedou všechny důležité kontaktní údaje na člena i rodiče a informace o zdravotním stavu, užívaných lécích a alergiích. Po přijetí se člen může účastnit schůzek i jednorázových akcí v rámci své družiny.

V současnosti neexistuje žádná centrální evidence členů, ze které by bylo možné snadno zjistit potřebných údajů o členech. Údaje jsou v papírové formě na přihláškách členů a každý družinový vedoucí má jen seznam kontaktů na své členy a jejich rodiče.

2.2.2 Placení členských příspěvků

Členské příspěvky se platí vždy jednou ročně a jejich výše může být každý rok jiná a liší se také její sazba pro děti a pro vedoucí.

Příspěvky aktuálně vybírají od dětí přímo vedoucí a předávají je oddílové účetní. Pro tento proces neexistuje zatím žádný centrální postup a každý družinový vedoucí si vybírání i předávání peněz se seznamem platících členů organizuje sám.

Příspěvky je možné po dohodě s účetní zaplatit i převodem na účet. Účetní poté ručně kontroluje od koho platba přišla.

2.2.3 Uskutečnění jednorázové akce

Jednorázové akce jsou většinou výpravy a tábor. Akce vzniká jejím naplánováním vedoucími. Dále jsou pozvaným členům, jejich rodičům i ostatním vedoucím rozeslány emaily s informacemi o plánované akci a s pozvánkou.

Pro přihlašování dětí na akce existuje sdílená tabulka viz obrázek 2.1. V tabulce je sloupec pro každou plánovanou akci ve které členové nebo jejich rodiče potvrdí nebo odmítnou účast. Obdobná tabulka existuje i pro přihlašování vedoucích.

Organizátor akce si před začátkem akce vytiskne seznam účastníků i s důležitými kontaktními údaji pro případ nějakých komplikací. V seznamu účastníků si poznamenává, kdo na akci skutečně dorazil a případně zda zaplatil.

2.2. Analýza současných procesů

	A	B	C	D	E	F	G	H	I
1		Jméno:	Hemžení	Výprava na srub	Semtětš	Vánoční srub	Noc plná her	Lyže	Vandř
2			14.-15.9.2018	12.-14.10.2018	23.-25.11.2018	14.-15.12.2018	26.1.2019	23.2.2019	22.-24.3.
3									
4			ne	ano	ne	ano	ano	ne	ne
5			ne	ne	ne	ano	ne	ne	ne
6			ano	ano	ano	ne	ano	ano, lyže	ano
7				ano přijedu v sobotu	ano	ano	ano	ano, lyže	
8									
9									ne
10					ne	ano	ano	ano	
11						ano - vlastní odvoz	ano		
12			ne	ne			ano		
13			ne				ne		ne
14									
15				ano		ano	ano	ne	ne
16						ano	ano		
17			ne	ano		ne	ano	ano, jen lyže	ne

Obrázek 2.1: Současná tabulka pro přihlašování na akce

Příprava seznamu účastníků včetně kontaktních údajů a vytváření pozvánky v grafickém editoru jsou časově náročné úkony, na které si vedoucí často stěžují. Dalším nedostatkem, který je komplikací pro děti i vedoucí je absence přehledného harmonogramu plánovaných akcí. V současnosti si každý musí nadcházející akce sám zaznamenávat do kalendáře nebo je poté musí zpětně hledat v doručených emailech.

2.2.4 Uskutečnění pravidelné schůzky

Na začátku každého školního roku se sejdou všichni aktivní vedoucí a dohadují harmonogram roku včetně toho v který den v týdnu a v které místnosti se budou scházet jednotlivé družiny. Tento harmonogram platí zpravidla celý školní rok přičemž schůzky se konají jen v ty dny, kdy se běžně chodí do školy, tedy v časech svátků a prázdnin schůzky zpravidla odpadají.

V některých družinách se na schůzkách některé aktivity budoují a stejné bodování se využívá i na některých výpravách nebo na táboře.

2.2.5 Rozesílání hromadných emailů

V současné době je pro rozesílání hromadných emailů nutné kontaktovat jediného vedoucího, který udržuje seznam emailových adres na členy oddílu. Problémem tedy je za prvé, že odesílání hromadných emailů je zbytečně zdlouhavé především ve chvílích, kdy je tento vedoucí nedostupný a za druhé, že není možné snadno vybrat seznam emailů jiné skupiny, než všechny děti a všechny vedoucí. Často by se totiž hodilo i další filtrování podle družin, účastníků nějaké akce, rodičů atp.

2.2.6 Sdílení fotografií z táborů a výprav

Na výpravách a na tábore je většinou alespoň jeden vedoucí, který pořizuje fotografie z akce. Po skončení akce předá flash disk s fotografiemi administrátorovi webu, který je ve volných chvílích nahrává na web.

Problém tohoto postupu je, že nahrávání fotografií je vždy závislé na jediném člověku a trvá až několik měsíců než se fotografie dostanou od fotografa na web.

2.3 Požadavky na nový systém

Na základě analýzy současných procesů a několika schůzek se zástupci organizace jsme rozhodli, že řešením by měl být informační systém, který bude splňovat následující požadavky. Požadavky uvedené v této kapitole byly tedy spoluvytvářeny a především schváleny zástupci organizace a jsou tedy závazným podkladem pro následný návrh.

Požadavky jsou rozděleny na funkční a nefunkční. Funkčními požadavky jsou takové, které se zaměřují na věcný a problémový obsah systému. [2] Nefunkční požadavky se zaměřují na okolnosti řešení, jaká omezení jsou kladena na systémové služby. [2]

U každého požadavku vždy uvedu i jeho prioritu a zda je nutné ho implementovat hned v první verzi aplikace, nebo může být součástí až některého z budoucích rozšíření. Priorita požadavku může být vysoká, střední nebo malá.

2.3.1 Funkční požadavky

2.3.1.1 F1 - Agenda akcí(výpravy, tábory a jiné akce)

U akcí bude možné vyplnit následující údaje: Název akce, Čas konání (od,do), uzávěrku přihlášek, obecný popis, místo srazu, místo návratu, cenu, seznam věcí.

Organizátor bude mít možnost na akci hromadně pozvat různé uživatele a nastavit kteří uživatelé budou organizátory akce a budou mít tedy i práva na úpravu této události. Pro výběr pozvaných členů je možné používat přednastavené seznamy často používaných skupin uživatelů (děti, vedoucí, dospělí členové, rodiče, členové konkrétní družiny). U organizátorů je možné textově zadat jakou roli mají na akci plnit.

Systém musí umožňovat vedoucím zakládat a editovat akce. Při vytváření by mělo být možné i automatické vygenerování pozvánky v PDF. Vzhled pozvánky budou vedoucí volit ze dvou šablon. Systém ne-umožní měnit ani stávající ani přidávat další šablony.

Děti i vedoucí by se měli přihlašovat na akce přes informační systém. Nezletilé děti přihlašují jejich zákonní zástupci. Nezletilé děti mohou mít vlastní přístup do systému, ale pokud se pokusí něco upravit nebo se přihlásit na

akci, tak se jejich zákonným zástupcům odešle email s požadavkem na schválení nebo zamítnutí provedené akce. Zletilé osoby se přihlašují sami. Zletilí si mohou zvolit, zda jejich příbuzní budou mít stejná práva k jejich profilu jako oni sami.

Organizátor akce si může vygenerovat tabulku se seznamem účastníků, která bude obsahovat také kontaktní údaje na rodiče a sloupečky pro odškrtnutí účasti a zaplacení.

Tento požadavek má vysokou prioritu a má být součástí první verze aplikace.

2.3.1.2 F2 - Evidence členů

Systém bude využíván členy, zákonnými zástupci členů, vedoucími, zdravotníkem a oddílovou účetní. Každý uživatel tedy může mít jednu i více následujících rolí.

- Člen nezletilý do 15 let – Práva jen na zobrazování, všechny pokusy o úpravy nebo přihlašování na akce jsou odeslány rodičům ke schválení.
- Člen nezletilý od 15 do 18 let – Stejná práva jako člen nezletilý do 15 let. Se souhlasem rodičů může nabýt stejných práv jako člen zletilý.
- Člen zletilý – Práva na zobrazování i úpravu informací o sobě a právo přihlašovat se sám na akce. Může si zvolit, zda jeho příbuzní budou mít právo zobrazovat stejné informace jako člen sám.
- Člen vedoucí – Všechna práva zletilého člena a navíc práva vytváření a úprava akcí (schůzky, výpravy, tábory a jiné události), přístup k informacím o všech uživateli (člen, zákonný zástupce, zdravotník, účetní)
- Zákonný zástupce – Právo jednat jménem nezletilého člena, případně i právo zobrazovat informace o zletilém členovi, pokud k tomu zletilý člen svolí.
- Zdravotník – Přístup k informacím o všech členech a jejich zákonných zástupcích.
- Účetní – Informace o placení příspěvků o všech členech.

Systém neumožní změnu práv jednotlivých rolí ani vytváření dalších rolí.

O každém uživateli evidujeme jeho přihlašovací a kontaktní údaje (jméno, příjmení, email, telefon). O členech evidujeme navíc adresu, datum narození, alergie, užívané léky, zdravotní stav, kontaktní údaje na zákonné zástupce a příslušnost ke družině. U vedoucího, účetní a zdravotníka evidujeme případné absolvované kurzy související s jejich rolí a u zákonných zástupců, kteří k tomu svolí evidujeme zaměstnání.

Požadavek na evidenci členů má vysokou prioritu a má být součástí první verze aplikace.

2.3.1.3 F3 - Evidovat placení členských příspěvků

Systém umožní evidovat placení členských příspěvků. Systém umožní členům vygenerovat si platební údaje pro platbu převodem a taková platba je po přijetí na účet automaticky spárována s členem a je mu prodlouženo členství. Oddílová účetní může zadávat informace o zaplacení v hotovosti. Vedoucí mohou do systému také zadat, že přijali platbu od dětí, v tu chvíli je stav platby nastaven na "peníze na cestě". Účetní může změnit stav platby na „zaplacené“ v momentě, kdy hotovost vloží do pokladny.

Priorita tohoto požadavku je střední. Alespoň přijímání hotovostních plateb by mělo být implementované již v první verzi systému.

2.3.1.4 F4 - Hromadné rozesílání emailů

Rozesílání hromadných emailů členům. Filtrování osob podle přednastavených seznamů (dětí, vedoucí, dospělí členové, rodiče, členové konkrétní družiny), ale i možnost přidat nebo odebrat uživatele ze seznamu příjemců ručně. Mohou provádět jen vedoucí. Text zprávy musí být možné zadávat prostřednictvím jednoduchého WYSIWYG editoru (při úpravě se zobrazuje přímo zformátovaný text). K emailu musí být možné připojit i soubory příloh (max. 10).

Po odeslání se příjemcům jednak odešlou emaily, ale také se jejich obsah zobrazí jako nepřečtená zpráva jakmile se přihlásí do systému. Uživatelé si v systému mohou zobrazit seznam všech odeslaných i přijatých zpráv a zvýrazní ty, které si ještě v systému nezobrazili. Emaily systém jen odešle a dále už se nestará, zda byli doručeny přečteny atp.

Požadavek F4 má střední prioritu a měl by být implementován již v první verzi systému.

2.3.1.5 F5 - Veřejná sekce webu

Nepřihlášení návštěvníci uvidí jen veřejnou prezentaci organizace členěnou stejně jako současný web <http://oregon.cx/>. Texty ve veřejné sekci se moc často nemění a proto bude pro začátek dostačující, když se budou upravovat přímo změnou souboru s jejich obsahem na serveru. V systému nebude možné změnit členění veřejné sekce ani přidávání dalších stránek.

Požadavek F5 má střední prioritu a měl by být implementován již v první verzi systému.

2.3.1.6 F6 - Foto galerie

Galerie fotografií z výprav, táborů a dalších akcí. Fotografie z jednotlivých akcí budou ve vlastní galerii a každá galerie bude spadat do jedné z kategorií jako tomu je současně (Pravěk, Výpravy, Tábory, Oregon dětem, Prckové).

Galerie mohou být veřejné nebo soukromé. Veřejnou galerii si mohou zobrazit i nepřihlášení uživatelé, ale soukromou galerii vidí jen přihlášení členové.

Přidávat fotografie mohou jen vedoucí. Systém umožní i hromadné nahrávání více fotografií najednou, ale nebude umožňovat jiné úpravy vkládaných fotografií než je jejich automatické zmenšení na maximální šířku 1200px a výšku 800px.

Požadavek F6 má malou prioritu a může být implementován až při pozdějším rozšíření systému.

2.3.1.7 F7 - Agenda schůzek

Zpravidla jednou ročně je nutné naplánovat schůzky jednotlivých družin.

Systém tedy umožní založit akce pro všechny schůzky na celý školní rok. Schůzky stejné družiny jsou vždy ve stejný den v týdnu a nekonají se o prázdninách ani státních svátcích. Vedoucí může v průběhu roku kteroukoliv schůzku zrušit.

Tento požadavek má malou prioritu a může být implementován až při pozdějším rozšíření systému.

2.3.2 Nefunkční požadavky

2.3.2.1 N1 - Responzivní design

Systém by se měl přehledně zobrazovat i na chytrých mobilních zařízeních s rozlišením alespoň (320x480px). Responzivní design není nutný v administrativní sekci.

2.3.2.2 N2 - Snadná použitelnost

Systém musí být snadno použitelný i pro běžné uživatele, kteří umí používat internetový prohlížeč k běžnému procházení internetu.

Rešerše existujících řešení

V následující kapitole provedu průzkum existujících řešení, která by mohla splňovat stanovené požadavky. Nakonec vyhodnotím, zda je vhodnější využít nebo rozšířit některé z existujících řešení nebo vyvinout nový informační systém na míru stanoveným požadavkům.

3.1 Hotový informační systém

Na trhu se v současnosti nachází spousta hotových informačních systémů specializovaných na určitou oblast. Řešení šitá na míru dětským klubům tohoto typu však nejsou běžně dostupná a proto se zde zaměřím i na informační systémy, které by alespoň částečně splňovaly požadované vlastnosti.

Ze všech prozkoumaných hotových informačních systémů se zdají být nejpoužitelnější takzvané „systémy pro školní administrativu“. Jedná se o informační systémy, které využívají různými vzdělávacími organizacemi k ukládání různých informací důležitých pro jejich činnost a komunikaci v rámci organizace, ale i s rodiči atp. Systémy typicky podporuje následující funkce:

- evidence žáků,
- tisk vysvědčení,
- tvorba rozvrhů,
- suplování,
- přijímací řízení,
- správa knihovny,
- plán akcí,
- docházkový systém ...

Mezi takové systémy patří například Bakaláři, iŠkola nebo Etřídnice. Za využívání se platí zpravidla roční licenční poplatky, který se pohybuje okolo 4000 Kč. Přizpůsobení systému je možné jen v rámci nastavení jednotlivých modulů a není možné systém upravovat ani rozšiřovat o další funkce – nedovoluje to licence ani fakt, že nemáme přístup ke zdrojovým kódům.[3] [4] [5]

3.2 Samostatné nástroje

Další možností je využití kombinace různých samostatných nástrojů zaměřených vždy na konkrétní požadavek nebo skupinu požadavků. Mezi takové nástroje určitě patří:

- Google Drive,
- Excel tabulka,
- Google kalendář,
- Facebook události,
- email

Využití těchto nástrojů je snadné a některé problémy se takto skutečně dají řešit, ale spousta požadavků se takto vyřešit nedá. Problémové je také to, že vznikají ne-konzistence v datech napříč nástroji.

3.3 Redakční systémy

Existují také nástroje a služby, které umožňují snadněji a rychleji vytvořit jak jednoduchý web, tak i složitější informační systém. Takové nástroje se většinou označují zkratkou CMS (content management system) nebo také česky jako redakční systémy. Redakční systémy jsou tedy nástroje, ve kterých lze s pomocí připravených šablon a rozšiřujících modulů vytvořit i složitý informační systém, který může nějakým způsobem podporovat i většinu našich požadavků. K využití redakčních systémů není nutné ani umět programovat, instalace a nastavování modulů se provádí většinou v grafickém rozhraní.[6]

Nevýhodou je, že pro splnění všech našich požadavků na systém se neobejdeme bez úpravy a doprogramování dalších funkcionalit. Některé redakční systémy vyžadují pravidelné poplatky ale i spousta kvalitních je poskytována zdarma. Často se však vyskytují problémy se zabezpečením.

3.4 Vyhodnocení rešerše

Po zvážení všech pro a proti a s přihlédnutím na mé vlastní zkušenosti s programováním podobně rozsáhlých systémů jsem se rozhodl pro vývoj vlastního informačního systému na míru s využitím PHP frameworku.

Vývoj vlastního informačního systému vyžaduje značné počáteční investice především do času programátora. Výhodou je, že výsledný systém bude splňovat všechny požadavky a nebude obsahovat nevyužívanou funkcionalitu, která by zhoršovala přehlednost a použitelnost jako u některých z existujících řešení.

Při snaze snížit náklady je vhodné pro implementaci vlastního informačního systému využít některý z PHP frameworků, které usnadní a zrychlí vývoj s využitím připraveného jádra systému a možností využívat hotové komponenty pro běžné problémy. Další výhodou je, že náklady na provoz takového systému spočívají pouze v pronájmu hostingu a domény, není třeba platit žádné pravidelné licenční poplatky.

Návrh a realizace systému

V následujících kapitolách popíši návrh a realizaci nového informačního systému pro tábornický oddíl Oregon. V návrhu budu vycházet z předchozích kapitol o analýze a navrhnu tedy nový informační systém, který následně i implementuji s využitím některého z PHP frameworků.

4.1 Metodika vývoje

Pro vývoj tohoto projektu, ale především pro jeho návazné rozšiřování bude využíváno agilní metodiky vývoje, která se jmenuje extrémní programování. Tato metodika se vyznačuje následujícími principy:

- iterativní vývoj,
- uživatel je součástí vývojového týmu,
- okamžité testování,
- práce ve dvojicích,
- neustálý refaktoring.

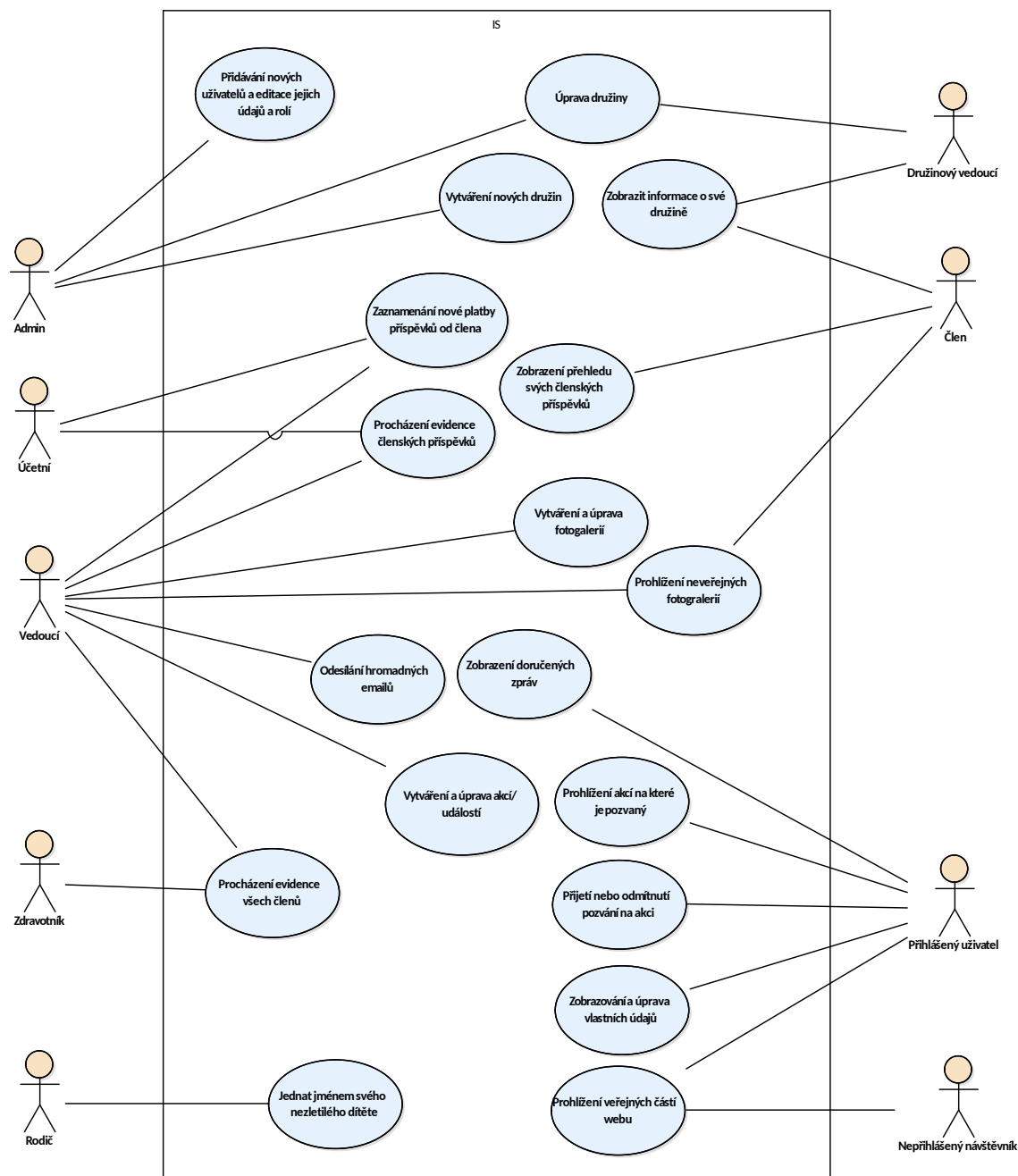
[7]

Tuto metodiku jsem zvolil především kvůli omezenému rozpočtu a proto, že další vývoj budou provádět vývojáři, kteří jsou přímo členy této organizace a mají možnost iterativně reagovat na aktuální stav.

4.2 Případy užití

Diagram 4.1 popisuje případy užití nového informačního systému včetně jejich aktérů. Případy užití pokrývají veškerou funkcionalitu, která bude následně implementována. Případy užití jsou užitečné i jako podklad pro závěrečné uživatelské testování.

4. NÁVRH A REALIZACE SYSTÉMU



Obrázek 4.1: Diagram zachycující případy užití a jejich aktéry

4.3 Použité technologie

Moje zkušenosti s některými konkrétními technologiemi ovlivnili i finální výběr použitých technologií pro tento projekt. Využití známých technologií je výhodné kvůli rychlejšímu vývoji a tedy i nižším nákladům než v případě nových technologií, které by vyžadovaly další studium. Škála technologií, se kterými mám zkušenosti je naštěstí dostatečně pestrá aby efektivně pokryla všechny požadavky.

V následujících kapitolách popíši hlavní technologie, které při realizaci projektu využiji.

4.3.1 PHP

PHP, neboli PHP: Hypertext Preprocessor je v současnosti jeden z nejpoužívanějších programovacích jazyků pro vývoj webových aplikací. PHP je interpretovaný jazyk, který se zpracovává na straně serveru a jeho hlavním cílem je komunikace s databází a zpracovávání dat, aby je bylo možné dynamicky zobrazovat do webové aplikace. Interpret jazyka PHP lze zprovoznit téměř na všech operačních systémech. [8]

Důležitá je, i podpora objektově orientovaného programování (OOP) a fakt, že PHP lze stáhnout a využívat zcela zdarma.

4.3.2 HTML a CSS

HTML je zkratka pro HyperText Markup Language, neboli hypertextový značkovací jazyk. HTML slouží k definování významu obsahu, zatímco CSS (Cascading Style Sheets) definuje vzhled stránky. Kód obou jazyků je ukládán v textové podobě na serveru a internetové prohlížeče je v grafické podobě zobrazují.[9]

4.3.3 Nette framework

Nette je český open source framework, který se skládá z mnoha samostatně použitelných komponent pro PHP. Umožňuje rychlý vývoj kvalitních a bezpečných aplikací. Připravené komponenty a postupy umožňují psát méně kódu a vytvářet snadno rozšiřitelné aplikace.[10]

Následně zmíním několik komponent Nette, které plánuji využívat:

- Latte - bezpečný šablonovací systém,
- Tracy - nástroj pro ladění PHP kódu,
- Dependency Injection - chytré předávání závislostí,
- Forms - vytváření a zpracovávání webových formulářů,
- RobotLoader - automatické načítání tříd a knihoven.

4.3.4 Bootstrap 4

Bootstrap 4 je open source sada nástrojů pro vývoj pomocí HTML, CSS a JS. Umožňuje využít před-připravené a vzhledově jednotné prvky snadno použitelné pro vlastní grafické uživatelské rozhraní (GUI).[11]

4.3.5 MySQL

Pro ukládání dat aplikace jsem zvolil relační databázovou technologii MySQL. Jedná se o rozšířenou technologii, kterou lze využívat zdarma a pro účely tohoto projektu bude postačující.

4.3.6 Doctrine

Doctrine je framework pro objektově relační mapování (ORM) v PHP. ORM je technika, která slouží k namapování objektů OOP na relační databáze. Výhodou je, že v aplikaci můžete s daty pracovat jako s entitami objektů, přestože se ukládají a získávají z běžné relační databáze.[12]

4.4 Architektura systému

Již v předchozích kapitolách jsem zmínil, že pro implementaci jsem zvolil PHP framework Nette a z toho vyplývá i softwarová architektura použitá pro tento projekt.

4.4.1 Model-View-Controller

Jedná se o architekturu Model-View-Controller (MVC), která se soustředí na oddělení kódu obsluhy, kódu aplikační logiky a kód zobrazujícího data. Takové rozdělení aplikaci zpřehledňuje, usnadňuje další rozvoj a umožňuje testování jednotlivých částí zvlášť.[13]

4.4.1.1 Model

Modely obsahují aplikační logiku systému. Modely neví o existenci kontrolerů ani view a navenek nabízí jen své pevné rozhraní.[13]

Já v modelové vrstvě využívám návrhový vzor fasáda pro třídy které zastřešují funkcionalitu nad jednou či více třídami entit. Každá fasáda implementuje rozhraní a pro komunikaci s ostatními částmi aplikace je využíváno výhradně těchto rozhraní. Mnou použitá implementace fasád využívá frameworku Doctrine pro namapování objektů entit na relační databázi, ale tuto implementaci modelové vrstvy je možné nahradit jakoukoliv jinou a při dodržení stanoveného rozhraní to nebude vyžadovat žádný zásah do ostatních částí systému.

```
<?php
namespace App\Model\Facades;
use App\Model\Entities\Member;

/**
 * Rozhraní fasády pro práci s členy.
 */
interface IMemberFacade
{
    /**
     * Najde a vrátí člena podle jeho ID.
     * @param int $id
     * @return Member|NULL
     */
    public function getMember($id);

    /**
     * Uloží nového člena.
     * @param Member $member
     * @return Member
     */
    public function insertMember(Member $member);
    ...
}
```

Výpis kódu 1: Ukázka rozhraní fasády v modelové vrstvě

4.4.1.2 View

Vrstva view, neboli pohled má na starosti zobrazování výstupu uživateli. Nette v této vrstvě využívá šablonovací systém Latte, který kombinuje prvky HTML se svými makry aby bylo snadné definovat jak a kde se má zobrazit dodaný výstup.[13]

4.4.1.3 Controller

Controler zpracovává požadavky uživatele, pomocí rozhraní modelové vrstvy posílá požadavky na aplikační logiku a poté požádá view o zobrazení výsledku. V Nette frameworku zastávají funkci controlerů třídy zvané presentery.[13]

4.5 Databázová struktura

Databáze má celkem 23 tabulek. Diagram zachycující kompletní databázovou strukturu je kvůli své velikosti umístěn v příloze B.

```

{form editParamedicForm}
  <div class="form-errors" n:if="$form->hasErrors()">
    <div class="alert alert-danger" n:foreach="$form->errors as $error">
      {$error}
    </div>
  </div>
  <div class="form-group">
    {label courses /}
    {input courses, class=>'form-control'}
  </div>
  {input send, class=>'btn btn-success'}
{/form}

```

Výpis kódu 2: Ukázka latte šablony ve vrstvě view



Obrázek 4.2: Ukázka GUI - menu ve veřejné sekci webu

4.6 Grafické uživatelské rozhraní

Při návrhu grafického uživatelského rozhraní (GUI) jsem se musel zaměřit především na splnění požadavků na responzivní design (požadavek N1) a na snadnou použitelnost (požadavek N2). Současně bylo třeba GUI vyřešit pokud možno jednoduše a levně, aby zbytečně nenarůstaly náklady.

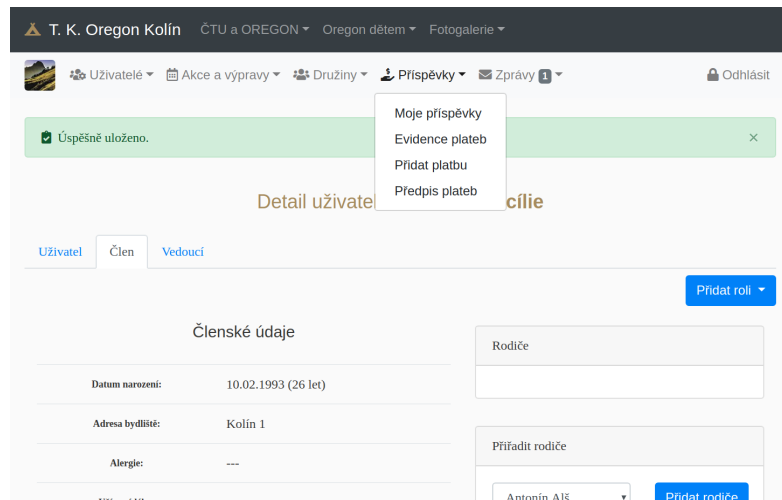
Zvolil jsem tedy nástroj Bootstrap 4, který umožňuje snadné vytváření uživatelského rozhraní a podporuje i responzivitu. Jednotlivé prvky tohoto nástroje jsou graficky stejnorodé, uživatelům internetu dobře známé a jejich ovládání je intuitivní. To všechno platí samozřejmě jen pokud je nástroj používán správně a s citem.

Velkou pozornost jsem musel věnovat návrhu navigačních prvků (menu). Pro menu veřejné sekce jsem zvolil formu horizontálního panelu s výrazným, téměř inverzním pozadím, který můžete vidět na obrázku 4.2.

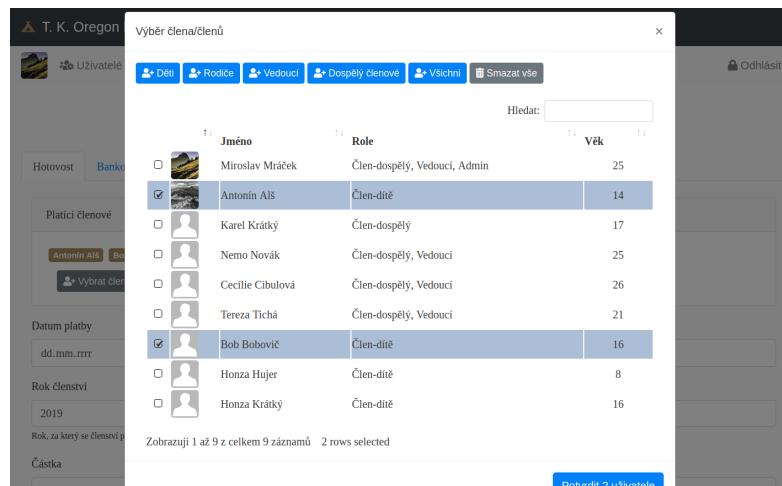
Pro přihlášení uživatele se pod současné menu přidá další lišta s položkami, které se liší podle rolí, které uživatel zastává. Navigační menu přihlášeného uživatele a další prvky grafického návrhu můžete vidět na obrázku 4.3.

Obrázek 4.5 zachycuje jak se vzhled stránky přizpůsobí displeji mobilního zařízení. Responzivita funguje dobře na všech veřejných stránkách, ale i na většině stránek v uživatelské sekci. Několik administrativních stránek s velkými tabulkami nebylo možné optimalizovat, ale to je v souladu s požadavky.

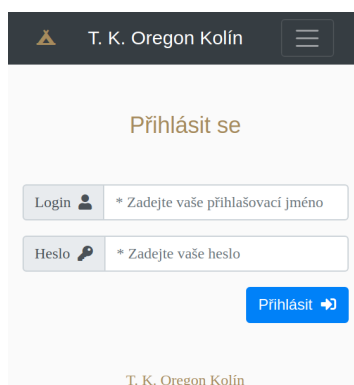
4.6. Grafické uživatelské rozhraní



Obrázek 4.3: Ukázka GUI - menu po přihlášení



Obrázek 4.4: Ukázka GUI - formulář pro zadávání platby příspěvků



Obrázek 4.5: Ukázka GUI - přihlašovací stránka na mobilním zařízení

4.7 Zefektivněné procesy

V této kapitole popíšeme některé procesy, které se podařilo s využitím nového IS zefektivnit.

4.7.1 Přijetí nového člena

První část procesu se nemění, tedy rodiče si od družinového vedoucího zjistí podrobnější informace a vedoucí rozhodnou, zda bude člen přijat. Pokud je rozhodnuto, že člen bude přijat, tak vedoucí s rolí administrátor vytvoří nové uživatelské účty pro člena i jeho zákonné zástupce a systém odešle každému uživateli uvítací email s instrukcemi a přihlašovacími údaji. Rodiče v systému vyplní a udržují aktuální důležité údaje o adresách, zdravotním stavu a alergiích člena. Všechny údaje jsou tedy v systému, kde k nim mají snadný přístup všichni vedoucí i zdravotníci.

4.7.2 Placení členských příspěvků

Účetní jednou ročně vytvoří ceník příspěvků pro daný rok, kde specifikuje cenu příspěvků za dítě, vedoucího a dospělého člena, který není aktivním vedoucím. Vytvořením ceníku vzniká automaticky všem členům požadavek na zaplacení příspěvků. Následuje rozeslání hromadného emailu (viz proces Rozeslání hromadných emailů), ve kterém si zaplacení příspěvků po všech členech vyžádá.

Člen po přihlášení k účtu vidí, že nemá zaplacené příspěvky a k zaplacení může zvolit platbu převodem podle zobrazených platebních údajů, nebo platbu v hotovosti. Pokud družinový vedoucí přijme platbu příspěvků od členů své

družiny, tak do systém tuto skutečnost zadá příslušným formulářem (viz obrázek 4.4), taková platba je ve stavu „peníze na cestě“, dokud nejsou peníze předány účetní, která změní stav platby na zaplacené.

Účetní i všichni vedoucí si mohou zobrazovat celou evidenci plateb. Včasné nezaplacení příspěvků některým členem je řešeno individuálně a v extrémním případě může vést k vyloučení člena, kdy dojde i k zablokování jeho účtu administrátorem.

4.7.3 Uskutečnění jednorázové akce

Jakmile vedoucí naplánuje výpravu, tak ji zadá do systému, kde vyplní všechny podstatné informace pro danou akci a vybere pozvané členy a další organizátory. Může využít i komponentu pro automatické generování pozvánky, která ze zadaných údajů vytvoří PDF pozvánku dle standardní šablony. Následuje rozeslání hromadného emailu s pozvánkou na akci.

Členové si mohou zobrazit seznam akcí na které jsou pozvaní a pokud ještě neskončil deadline na přihlašování a nebyla vyčerpána kapacita akce, tak mohou jedním kliknutím přijmout nebo odmítnout účast, případně ke své účasti uvést poznámku.

Organizátoři akce si mohou zobrazit počet i seznam účastníků a jedním kliknutím vygenerovat export s tabulkou účastníků a kontaktními údaji, který si mohou před začátkem akce vytisknout.

4.7.4 Rozesílání hromadných emailů

Všichni vedoucí mohou rozesílat hromadné emaily všem uživatelům informačního systému. Mohou ručně zaškrtnávat příjemce, nebo hromadně celé skupiny uživatelů (děti, rodiče, vedoucí, dospělí členové, členové konkrétní družiny nebo všichni). Text zprávy může snadno formátovat pomocí jednoduchého WISIWYG (What you see is what you get) textovém editoru. K emailu je možné přiložit až 3 soubory s přílohami. Vytvořenou zprávu může vedoucí odeslat buď ihned, nebo naplánovat její odeslání na pozdější termín.

Zprávy rozeslané pomocí IS jsou jednak odeslány jako běžné emaily, ale také se zobrazují všem příjemcům přímo v systému.

4.7.5 Sdílení fotografií z táborů a výprav

Všichni vedoucí mají právo vytvářet v systému nové galerie fotografií a plnit je fotografiemi z výprav a akcí. Galerie mohou označovat jako soukromé, které se zobrazují jen přihlášeným členům, nebo jako veřejné, které vidí všichni návštěvníci veřejné části webu.

Fotografie se před uložením automaticky zmenší na maximální velikost 1200x800px, protože při prohlížení fotografií na počítači, nebo mobilním zařízení je to naprosto dostatečné, fotografie se načítají rychleji a nezabírají příliš místa na disku.

4. NÁVRH A REALIZACE SYSTÉMU



Obrázek 4.6: Screenshoty používání ladícího nástroje Tracy

4.8 Testování vývojářem

Testování mnou jakožto vývojářem probíhalo v průběhu celého vývoje vždy po dokončení každé části aplikace. Pro tyto účely jsem využíval na mém notebooku lokálně nainstalovaný PHP a MySQL server z balíčku XAMPP. Pro každou funkcionalitu jsem si vytvořil malou sadu testovacích dat. Zaměřoval jsem se na to, aby data testovala pozitivní i negativní testy, tedy jestli systém umožňuje provádět požadovanou funkci, ale i jestli se chová správně při zadávání nesmyslných dat nebo při pokusu o neoprávněný přístup.

Toto testování odhalovalo pravidelně velké množství syntaktických i logických chyb, a při jejich odhalování byla velice nápomocná i komponenta Tracy (obrázek 4.6), díky které se většina chyb dala rychle a přesně identifikovat.

Chyby odhalené v tomto testování jsem opravoval ihned, protože jejich oprava je jednodušší a rychlejší, dokud mám danou problematiku v čerstvě paměti. Chyby je možné odhalit i v dalších fázích testování a opravit později, ale to je už několikanásobně nákladnější.

4.9 Nasazení

Po dokončení implementace jsem přistoupil k nasazení na webhostingový server. Pro nasazení byl zvolen webhosting Ebola BASIC, který podporuje všechny

použité technologie a má velmi dobrou dostupnost 99.999%. Následně uvedu výčet některých důležitých vlastností tohoto webhostingu:

- platforma Linux, Apache webserver
- htaccess, mod_rewrite, SSI, WebDAV
- podpora pro PHP 5.2 – 7.3
- Memory_limit 128 MB
- Max_execution_time 90s
- Max_input_time 90s
- Post_max_size až 100MB
- Upload_max_filesize až 100MB
- celkový diskový prostor 10GB
- databáze MySQL 5.7
- maximální velikost databáze je 50MB
- maximální počet databází je 20
- neomezené množství přenesených dat
- FTP/s, POP3/s, IMAP/s, SMTP/s
- Neomezený počet CRON úloh
- minimální frekvence CRON úloh je 30 min

[14]

Na webhostingu jsem nejprve vytvořil novou databázi a nahrál do ní prázdnou databázovou strukturu. Přes FTP server jsem nahrál všechny potřebné soubory a upravil konfigurační soubor config.local.neon, tak aby se připojoval ke správné databázi. Tím byl nový IS úspěšně spuštěn na testovací doméně a ve spolupráci s dalšími vedoucími z organizace jsme systémem začali plnit daty o reálných uživateli.

Pro účely uživatelského testování a v úvodní fázi, než se odladí nedostatky a uživatelé se dostatečně seznámí s novým systémem, bude nový systém fungovat na testovací doméně (www.oregon.mirocloud.cz) a původní web na doméně (www.oregon.cx) zůstane stále funkční.

Uživatelské testování

Pro účely uživatelského testování jsem si připravil několik testovacích scénářů zaměřených na nejdůležitější a nejčastěji prováděné případy užití. V této kapitole vám podrobněji popíši průběh uživatelského testování a podrobněji popíši 2 konkrétní test scripty.

Jako testery jsem zvolil reálné uživatele systému, konkrétně 2 vedoucí z této organizace. Testy byly prováděny především na běžném kancelářském notebooku a některé i na mobilním telefonu s operačním systémem Android a rozlišením obrazovky 480x854px.

Každý test probíhal tak, že uživatel seděl u počítače, na stole měl vytištěný test script s instrukcemi podle kterých postupoval a za ním jsem stál já abych sledoval jak test probíhá, zaznamenával si poznámky a případně uživateli poradil, pokud se zastaví na nějakém problému. Uživatel měl za úkol postupovat podle instrukcí, plnit zadané úkoly a nahlas komentovat co právě dělá a proč si myslí, že to je správně.

U většiny testů dochází k zásahům do databáze a proto bylo nutné před každým testem uvést systém do výchozího stavu, aby bylo možné test dokončit přesně podle instrukcí v test scriptu.

5.1 Výsledky testování

Uživatelské testování odhalilo množství případů neintuitivního nebo pro některé uživatele matoucího chování. V těchto případech se nejednalo přímo o chyby, ale spíše o vlastnosti systému, které nejsou pro nezaškoleného uživatele zcela intuitivní. Tyto nedostatky ukazují, že uživatelské testování je hodně důležité a pokud chceme vyvíjet intuitivní řešení, neobejdeme se bez testerů, kteří se nepodíleli na vývoji. Některé z těchto neintuitivních chování byly opraveny a ostatní budou řešeny v dalších verzích systému, pokud praxe ukáže, že to je skutečně třeba.

5. UŽIVATELSKÉ TESTOVÁNÍ

Test Script 1 - Vkládání nových uživatelů (vysoká priorita):

(Vžijte se do role administrátora, který se snaží vložit do systému nové uživatele.)

- Přihlašte se k administrátorskému účtu (login:admin, heslo:df8Fw6210).
- Vytvořte nové uživatelské účty a vyplňte základní údaje pro následující uživatele:

Login:	Heslo:	Jméno a příjmení:	Email:	Telefon:
roman	roman326	Roman Krátký	kratky@seznam.cz	721000001
romana	romana427	Romana Krátká	kratky@seznam.cz	721000002

- Oběma právě vytvořeným uživatelům přiřadte roli Rodič.
- Vytvořte nový uživatelský účet a vyplňte základní údaje pro následujícího člena (nezapomeňte přiřadit roli Člen):

Login:	Heslo:	Jméno a příjmení:	Email:	Telefon:	Narozen:	Rodiče:
honza	honza538	Honza Krátký	---	721000003	3.3.2003	Roman a Romana

- Vytvořte nový uživatelský účet a vyplňte základní údaje pro následujícího uživatele a přiřadte mu roli člen a vedoucí:

Login:	Heslo:	Jméno a příjmení:	Email:	Telefon:	Narozena:
tereza	tereza649	Tereza Tichá	tereza@ticha.cz	721000004	7.7.1997

- Přejděte na seznam uživatelů a zkontrolujte, zda jsou všichni výše zmínění uživatelé vytvořeni a že mají správně nastavené všechny zmíněné informace.
- Odhlašte se

Obrázek 5.1: Test script - Vložení nové platby příspěvku

Test Script 4 – Vložení nové platby příspěvku (vysoká priorita)

(Vžijte se do role účetní, který se snaží vložit do platbu příspěvků přijatou v hotovosti.)

- Přihlašte se jako účetní (login:ucetni, heslo:sdah6340).
- Zadejte do systému přijetí platby členských příspěvků za následující členy:

Jméno a příjmení:	Částka:	Datum přijetí platby:
Honza Krátký	500	dnas
Tereza Tichá	500	dnas

- Přejděte do evidence plateb a zkontrolujte, že oba členové mají platbu vloženu:
- Odhlašte se

Obrázek 5.2: Test script - Vložení nové platby příspěvku

Testování odhalilo i některé jasné chyby, jako například chybějící ověřování, že uživatelské jméno již existuje, dříve než dojde k pokusu o zápis do databáze a jiné. Všechny tyto chyby jsem opravil.

Vyhodnocení přínosů

6.1 Náklady na realizaci a provoz

Systém jsem navrhoval a vyvíjel jako svou bakalářskou práci a tedy bezplatně, ale i přesto je možné alespoň zhruba vyčíslit odhadované náklady na jeho vytvoření, pokud by byl vyvíjen placeným vývojářem. V tabulce 6.1

Tabulka 6.1: Rozpis časové náročnosti jednotlivých činností

Činnost	Počet strávených hodin
Analýza a návrh	26 h
Implementace	90 h
Uživatelské testování	4 h
Součet	120 h

Při hodinové sazbě 400 Kč/hod lze vývoj nového informačního systému vyčíslit na zhruba 48 000 Kč. Náklady na provoz systému budou zhruba stejné jako doposud a to maximálně 1 000 Kč ročně za webhosting a doménu.

6.2 Přínosy

Přínosem nového systému je, že se zefektivnil pět ze šesti klíčových procesů v této organizaci. Jedná se o procesy přijetí nového člena, placení členských příspěvků, uskutečnění jednorázové akce, rozesílání hromadných emailů a sdílení fotografií z táborů a výprav. Nová evidence členů, plateb, jednorázových akcí a snadné rozesílání emailů ušetří především vědoucím a účetní dohromady desítky hodin času ročně, které mohou věnovat prospěšnějším činnostem.

Přínosem oproti jiným řešením je i fakt, že je možné systém dále vylepšovat a postupně rozšiřovat o podporu dalších procesů.

Závěr

V této práci jsem popsal analýzu, návrh a realizaci nového informačního systému pro tábornický klub Oregon Kolín. Na základě analýzy organizace a rešerše existujících řešení jsem rozhodl, že nejvhodnějším řešením bude navrhnout nový informační systém přesně podle dohodnutých požadavků s využitím PHP frameworku Nette. Systém jsem navrhl, implementoval a otestoval uživatelskými testy do kterých byli zapojeni přímo reální uživatelé tohoto systému, tedy členové oddílu Oregon. Na závěr jsem provedl vyhodnocení přínosů a nákladů. Náklady na vývoj nového informačního systému byli odhadnuty na zhruba 48 000 Kč. Přínosem je, že se podařilo zefektivnit pět ze šesti klíčových procesů a výsledný systém je možné dále rozšiřovat.

Splnil jsem tedy všechny stanovené cíle a výstupem mé práce je první verze informačního systému, který splňuje všechny funkční požadavky s vysokou prioritou (F1,F2) a střední prioritou (F3,F4,F5) a navíc i jeden požadavek s nízkou prioritou (F6). Nefunkční požadavky byly splněny všechny. První verze informačního systému včetně dokumentace zdrojových kódů je k dispozici na přiloženém CD.

Systém je navrhnout a implementován tak, aby byl snadno rozšiřitelný a v jeho dalších verzích mohly být implementovány i ostatní funkce s nízkou prioritou a mohl být i nadále rozvíjen. S přihlédnutím na nutné úpravy by tento systém mohl být využitelný i některými dalšími podobnými organizacemi.

Vypracování takto rozsáhlého projektu mi přineslo mnoho cenných zkušeností. Těšilo mne vytvářet systém, který pomáhá dobré věci.

Bibliografie

1. *Tábornický klub OREGON Kolín* [online] [cit. 2019-04-15]. Dostupné z: <http://www.oregon.cx/>.
2. ŠARMANOVÁ, Jana. *Databázové a informační systémy*. VŠB, 2008.
3. *Bakaláři - ceník* [online] [cit. 2019-03-02]. Dostupné z: <https://www.bakalari.cz/Home/Prices>.
4. *iSkola - ceník* [online] [cit. 2019-03-02]. Dostupné z: <https://www.iskola.cz/iskola/cenik>.
5. *etridnice - ceník* [online] [cit. 2019-03-02]. Dostupné z: <https://www.etridnice.cz/ceny/>.
6. NOVÁK, Tomáš. *Bezpečnost open source redakčních systém se zaměřením na CMS Joomla*. 2013.
7. SOMMERVILLE, Ian. *Softwarové inženýrství*. Computer Press, Albatros Media as, 2017.
8. HOPKINS, Callum. *PHP okamžitě*. Computer Press, Albatros Media as, 2017.
9. CASTRO, Elizabeth; HYSLOP, Bruce. *HTML5 a CSS3*. Computer Press, Albatros Media as, 2017.
10. FOUNDATION, Nette. *Seznámení s Nette Frameworkem* [online]. 2008-2019 [cit. 2019-05-01]. Dostupné z: <https://doc.nette.org/cs/2.4/getting-started>.
11. *Bootstrap* [online] [cit. 2019-04-03]. Dostupné z: <https://getbootstrap.com/>.
12. TICHÝ, Jan. *Doctrine 2: úvod do systému*. *Zdroják. cz*. 2010.
13. FOUNDATION, Nette. *MVC aplikace & presentery* [online]. 2008-2019 [cit. 2019-05-01]. Dostupné z: <https://doc.nette.org/cs/2.4/presenters>.

BIBLIOGRAFIE

14. *eBola.cz - srovnání hostingových tarifů* [online] [cit. 2019-05-02]. Dostupné z: <https://ebola.cz/webhosting>.

Seznam použitých zkratk

- IS** Informační systém
- PDF** Portable Document Format
- CMS** Content management system
- OOP** Objektově orientované programování
- PHP** PHP: Hypertext Preprocessor
- CSS** Cascading Style Sheets
- HTML** HyperText Markup Language
- JS** JavaScript
- ORM** Object-relational mapping
- MVC** Model-view-controller
- WISIWYG** What you see is what you get

Schéma databáze

Obsah přiloženého CD

readme.txt	stručný popis obsahu CD
src	zdrojové kódy implementace
├─ app	zdrojové kódy aplikace (php)
├─ client_side	zdrojové kódy frontendu (js, css, less)
├─ www	index.php, obrázky, minifikované css a js
├─ db.sql	SQL script pro vytvoření struktury databáze
├─ readme.md	instrukce k sestavení a nasazení projektu
text	text práce
├─ thesis.pdf	text práce ve formátu PDF