



Zadání bakalářské práce

Název:	Web pěveckého sboru Gaudium Praha
Student:	Veronika Vlková
Vedoucí:	Ing. Jan Horáček
Studijní program:	Informatika
Obor / specializace:	Webové a softwarové inženýrství, zaměření Webové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	do konce letního semestru 2022/2023

Pokyny pro vypracování

Cílem práce je navrhnout a implementovat webovou aplikaci pro pěvecký sbor Gaudium Praha. Součástí aplikace bude rozhraní pro prezentaci své činnosti pro veřejnost a rozhraní pro interní potřeby sboru. Interní systém bude umožňovat sboru správu uživatelů, událostí, komunikaci se členy a snadnou editaci prezentační části aplikace.

Cíl práce:

- Analyzujte aktuální web sboru Gaudium (gaudiumpraha.org).
- Analyzujte alespoň tři weby s podobnou tematikou.
- Na základě komunikace s členy sboru analyzujte jejich nároky na web.
- Navrhněte architekturu a funkce webu tak, aby splňovala požadavky klienta.
- Aplikaci naimplementujte (frontend i backend).
- Proveďte nasazení a uživatelské testování webu.

Bakalářská práce

WEB PĚVECKÉHO SBORU GAUDIUM PRAHA

Veronika Vlková

Fakulta informačních technologií
Katedra softwarového inženýrství
Vedoucí: Ing. Jan Horáček
11. května 2023

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2023 Veronika Vlková. Odkaz na tuto práci.

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: Vlková Veronika. *Web pěveckého sboru Gaudium Praha*. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií, 2023.

Obsah

Poděkování	v
Prohlášení	vi
Abstrakt	vii
Seznam zkratek	viii
1 Úvod	1
1.1 Cíle práce	1
1.2 Struktura práce	1
2 Teoretická část	3
2.1 Analýza současného řešení	3
2.1.1 Vzhled	3
2.1.2 Jazyková adaptace	4
2.1.3 Interní informace	4
2.1.4 Přidávání a úprava obsahu	5
2.2 Analýza dalších webových stránek	6
2.2.1 Notre Dame ¹	6
2.2.2 Pueri Gaudentes ²	7
2.2.3 Pražská kantiléna ³	9
3 Praktická část	11
3.1 Požadavky na aplikaci	11
3.1.1 Funkční požadavky	11
3.1.2 Nefunkční požadavky	12
3.2 Zvolení systémů	13
3.3 Implementace aplikace	14
3.3.1 Zvolené systémy	15
3.4 Uživatelské testování	15
3.5 Ukázka aplikace	15
4 Závěr	19
4.1 Splnění cílů	19
4.2 Možnosti rozšíření	19
A Návod na spuštění a používání aplikace	21
A.1 Návod na spuštění aplikace	21
A.2 Návod k použití	21
Obsah přiloženého zip souboru	25

Seznam obrázků

2.1	Hlavní stránka webu Gaudium Praha, zdroj: [1]	4
2.2	Část stránky v inspektoru – formátování pomocí prázdných řádků, zdroj: [1] . . .	6
2.3	Hlavní stránka webu Notre Dame, zdroj: [8]	7
2.4	Posuvník na okraji stránky a na kraji boxu s obsahem, zdroj: [8]	7
2.5	Hlavní stránka sboru Pueri Gaudentes, zdroj: [10]	8
2.6	Menu jako obrázky na stránkách Pueri, zdroj: [10]	8
2.7	Zobrazení úvodní stránky Pražské kantilény na počítači, zdroj: [13]	9
2.8	Zobrazení úvodní stránky Pražské kantilény na mobilním telefonu, zdroj: [13] . .	10
3.1	Diagram zobrazující historii a množství programovacích jazyků, zdroj: http://rigaux.org/language-study/diagram.png	14
3.2	Hlavní stránka aplikace	15
3.3	Zobrazení zpráv na neveřejné stránce	15
3.4	Zobrazení skladeb	16
3.5	Zobrazení rozhraní vedení pro správu členů	16
3.6	Potvrzovací okénko před smazáním s možností návratu	16

Seznam výpisů kódu

3.1	Ukázka šablony Twig pro zobrazení příspěvků	17
-----	---	----

Chtěla bych poděkovat svému vedoucímu práce Ing. Janu Horáčkovi, který mi trpělivě odpovídal na mé dotazy a provázel mě vytvářením závěrečné práce. Dále bych chtěla poděkovat svým kamarádům Tomáši Hegerovi, Petrovi Adámkovi, Davidu Bernhauerovi, Mariusovi Solbergovi a Ghormoonovi za přínosné rady během psaní práce a za podporu během celého studia. Chtěla bych také poděkovat především svým rodičům, na které se vždy mohu spolehnout a v neposlední řadě mé poděkování patří i širší rodině a dalším přátelům, kteří mi byli v průběhu studia oporou, případně se také zúčastnili uživatelského testování.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracovala samostatně a že jsem uvedla veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při 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. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, 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 11. května 2023

.....

Abstrakt

Cílem této práce je vytvoření webové aplikace pro pěvecký sbor. Zaměřila jsem se především na co největší jednoduchost používání, což ocení především starší generace. K nalezení řešení byla použita analýza několika webových stránek jiných sborů. Řešení obsahuje použití frameworku Symfony, šablonovacího jazyka Twig a knihovny Doctrine pro přístup k databázi. Vytvořené řešení poskytuje jak možnost prezentace sboru široké veřejnosti, tak přihlášení uživatelů (členů sboru) a zobrazení neveřejných informací, včetně rozšiřujících informací k akcím viditelným pro veřejnost (např. čas srazu členů sboru). Práce také řeší jednoduché vkládání a základní úpravu příspěvků na veřejnou i neveřejnou webovou stránku.

Klíčová slova webová aplikace pro sbor, kalendář akcí, neveřejné informace, PHP, Symfony

Abstract

The aim of this thesis is to create a web application for a choir. I focused mainly on making it as easy to use as possible, which will be appreciated especially by the older generation. An analysis of several websites of other choirs was used to find a solution. The solution involves the use of the Symfony framework, the Twig templating language and the Doctrine library for database access. The developed solution provides the possibility to present the choir to the general public as well as allow users (choir members) to log in and display non-public information, including additional information on events visible to the public (e.g. time of choir members' meetings). The thesis also addresses simple insertion and basic editing of posts to the public and non-public web pages.

Keywords web application for choir, calendar of events, non-public information, PHP, Symfony

Seznam zkratk

alt	atribut v HTML, označuje alternativní text pro obrázek
CSS	Cascading Style Sheets
ctrl+f	klávesová zkratka používaná pro vyhledání obsahu v souboru/na stránce
GUI	Graphical User Interface
HTML	Hypertext Markup Language
JSP	JavaServer Pages
MIDI	Musical Instrument Digital Interface
MVC	Model, View, Controller
PHP	Hypertext Preprocesor (původně Personal Home Page)
W3C	World Wide Web Consortium

Kapitola 1

Úvod

Sborový zpěv je poměrně rozšířená zájmová činnost. Lidé zpívají profesionálně, ale i jen tak pro radost. Ovšem i při zpívání pro radost, pokud to je alespoň trochu organizovaná a ustálená skupina, je potřeba také určitá administrativa kolem.

Takový sbor postupem času potřebuje celkem dost věcí - například sdělení sbormistra členům, záznamy o plánovaných událostech, úložiště not, nebo přehled toho, kdo a kdy se může jaké akce zúčastnit. Řešení je mnoho - posílání e-mailů, různá úložiště jako např. Google disk a sdílené tabulky... Jenže do sboru nechodí jen mladí lidé, ale moc ráda si zazpívá i starší generace. A pro tu ovládnutí počítače může být leckdy náročné, nehledě na to, že se jim občas povede něco omylem smazat, přepsat, či další drobnosti. Nemluvě o tom, když ze starší generace jsou i sbormistři a s přidáváním obsahu na stránky někdy bojují.

Jedním takovým sborem je i smíšený pěvecký sbor Gaudium Praha. Jejich aktuální řešení je značně náchylné k nechtěným chybám a problémům, které jim akorát kazí radost z toho, co dělají. Proto bych tento problém ráda ve své práci vyřešila. Chtěla bych vytvořit webovou aplikaci, která by byla především jednoduchá a intuitivní i pro starší generaci, co nejvíce automatická a zároveň by ve velké míře zamezovala vzniku nezamýšlených chyb, ať už neopatrným ovládnutím, či španým podíváním se, který řádek tabulky má člověk upravit, když je mu umožněno zasahovat do všech...

1.1 Cíle práce

Cílem práce je analýza aktuálního řešení webových stránek a jeho nedostatků, analýza několika webových stránek jiných pěveckých sborů, a následně vytvoření a nasazení webové aplikace pro pěvecký sbor. Cílem aplikace je obsahovat jak prezentační část – pro veřejnost, tak neveřejnou část pro členy sboru. Ta zabírá většinu práce. Aplikace cílí především na jednoduchou ovladatelnost a intuitivnost i pro starší generaci. Dalším cílem je co nejvíce věcí zautomatizovat – provádět bez zásahu uživatelů (například využívat již jednou zadané informace na více místech – oznámení o koncertě jak ve veřejné části, tak v neveřejné části doplněné o další informace).

Dalším cílem je po nasazení aplikaci otestovat na reálných uživatelích a případně drobně upravit nedostatky aplikace na základě jejich připomínek.

1.2 Struktura práce

První kapitola seznamuje čtenáře s obsahem práce a uvádí jej do problematiky, vysvětluje, proč a čím je práce přínosná. Druhá kapitola se zaměřuje na nedostatky aktuálního řešení sboru Gaudium Praha a zkoumá weby dalších sborů. Ve třetí kapitole se čtenář podrobněji seznámí s řešením aplikace a čtvrtá zhodnotí splnění cílů a nastíní možná rozšíření.

Teoretická část

V této kapitole se autor zaměří na analýzu současných webových stránek, používaných systémů a jejich nedostatků. Analyzuje také webové stránky dalších sborů, minimálně jejich veřejnou část.

2.1 Analýza současného řešení

Současné řešení webu ¹ vypadá jako z období kolem přelomu tisíciletí, působí zastarale a ani pro osoby starající se o obsah není toto řešení pohodlné.

2.1.1 Vzhled

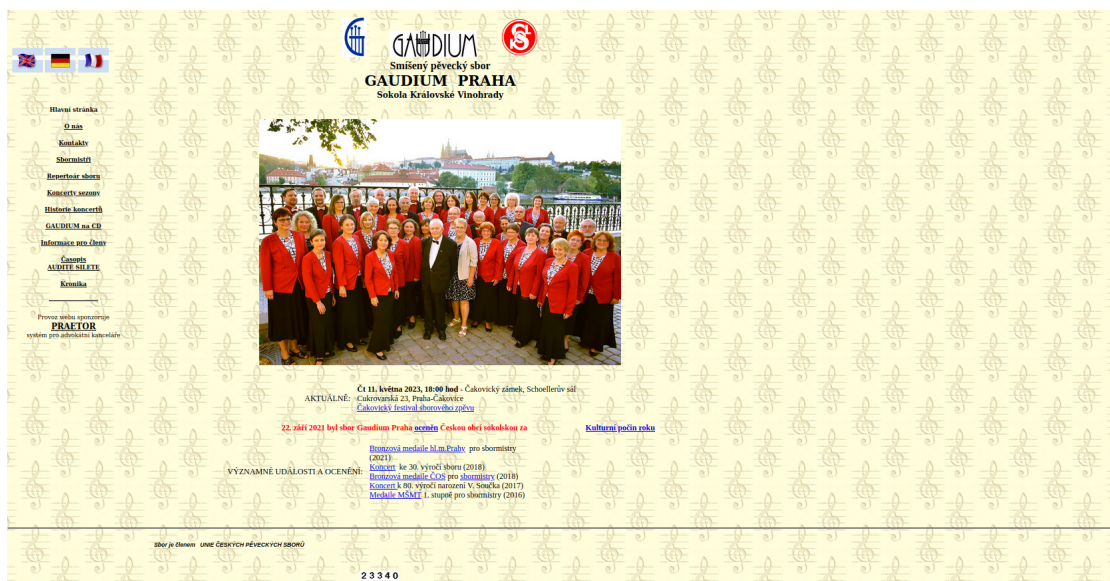
Jak už jsem zmínila výše, vizuální stránka nepůsobí příliš moderně, skoro až iniciuje otázku, zda jsou webové stránky ještě funkční a pravidelně aktualizované. (obr. 2.1) Přestože grafická stránka aplikace není něco, čím se primárně zabývám, pro celkovou funkčnost je to důležité. Další věcí k nápravě je zobrazení stránek např. na chytrém telefonu. Ve stávajícím řešení se při prohlížení stránek na mobilním telefonu nic nezmění, a jak uvádí i [2], od roku 2016 více lidí prohlíží stránky přes mobilní telefon (či tablet) než z běžného počítače (desktopu).

Responzivní web

„Za responzivní můžeme označit takové webové stránky, které jsou navrženy a realizovány s ohledem na použitelnost a přizpůsobitelnost různým rozlišením v různých zařízeních (stolní počítače, notebooky, netbooky, tablety nebo mobilní zařízení). Pojem Responzivní web design (RWD) představil již v květnu 2010 americký programátor Ethan Marcotte na blogu A LIST Apart, první implementace se objevila již v následujícím roce.“ [3]

Ethan Marcotte popisuje responzivní design jako cestu vpřed, umožňující zobrazení na různých zařízeních a zdůrazňuje důležitost přizpůsobení se. („Now more than ever, we’re designing work meant to be viewed along a gradient of different experiences. Responsive web design offers us a way forward, finally allowing us to “design for the ebb and flow of things.”“ [4])

¹<http://gaudiumpraha.org/> ([1])



■ Obrázek 2.1 Hlavní stránka webu Gaudium Praha, zdroj: [1]

2.1.2 Jazyková adaptace

Přestože Gaudium není výdělečná organizace, aplikace má sloužit i k prezentaci sboru, a tedy by měly zaujmout, například kvůli účasti na koncertech. Dle článku na blogu Weglot přispívá možnost lokalizace k zaujetí větší části populace a zároveň udržení její pozornosti. Lidé oceňují možnost si přečíst obsah v jejich rodném jazyce, případně alespoň jazyce, kterému rozumí (např. jedná-li se o cizince). [5]

Současné řešení sice zdánlivě nabízí možnost přepnutí do jiného jazyka, ale při bližším prozkoumání se nejedná o lokalizaci, ale o hypertextové odkazy na jiné stránky, což by nevedilo, ovšem toto řešení je náchylné k chybám a několik chyb tam také najdeme – např. z anglické verze se při pokusu o prohlédnutí si sborových CD dostaneme opět na českou verzi dané stránky a i celý zbytek webových stránek se přepne zpět do češtiny, včetně bočního menu, které bylo o krok zpět už přeložené. Daný uživatel je tak nucen znovu a znovu přepínat stránky do svého zvoleného jazyka, což je po pár cyklech tohoto procesu značně otravné.

2.1.3 Interní informace

Sbor potřebuje místo pro sdílení interních informací, ať už jsou to podrobnosti ke koncertu (jako např. čas srazu a oblečení), nebo jaké si vzít noty na další zkoušku. Do toho širší veřejnosti nic není. Některé informace je pak třeba chránit ze zákona, ať už se jedná o osobní informace, nebo třeba noty a některé nahrávky (tam je zas zákaz šíření z důvodu autorského zákona a dalších). V současném řešení je uživatelská sekce chráněna jménem a heslem, ovšem jméno i heslo je pro všechny stejné. A ani se moc často nemění. Znamená to, že kdokoliv byl jednou členem sboru, má i nadále přístup do neveřejné části (dokud se heslo nezmění, ale jak už jsem zmiňovala výše, to se neděje moc často). Jak zmiňuje University of Washington, sdílení účtu mezi více lidmi způsobuje hned několik bezpečnostních rizik, např. horší možnost zjištění a dokázání, kdo z daných uživatelů vykonal danou akci, nebo si zobrazil přístupná data. Dalším rizikem je vyšší šance kompromitování hesla a, jak už bylo zmíněno výše, složitější vyřazení určité osoby z možnosti přístupu k datům. [6]

Přihlašování se na akce

Důležitým prvkem v organizaci sboru je vědět s jakými členy lze kdy počítat. Proto je dobré mít možnost tyto informace zaznamenávat. Některé sbory to řeší např. napsáním se na papír, což ale přináší několik nevýhod, jako je nemožnost zaznamenání při absenci na zkoušce a nemožnost svou odpověď aktualizovat.

V Gaudiu řeší tuto problematiku sdílenou Google tabulkou². Vyřeší se tím sice dva výše zmíněné problémy, ale nastanou problémy jiné. Za zmínění stojí možnost upravovat jakýkoliv řádek, tedy nikoliv jen ten se svým jménem (možná je to sice nějakým způsobem řešitelné, ale to by zas vyžadovalo např. individuální Google účet pro každého, což je opět složitější). Chyba tedy může nastat ať už záměrně (což je sice nepravděpodobné, ale ta možnost existuje), nebo omylem (obzvláště při vyplňování na mobilu či lidmi se zhoršeným zrakem se člověk snadno strefí o řádek vedle).

2.1.4 Přidávání a úprava obsahu

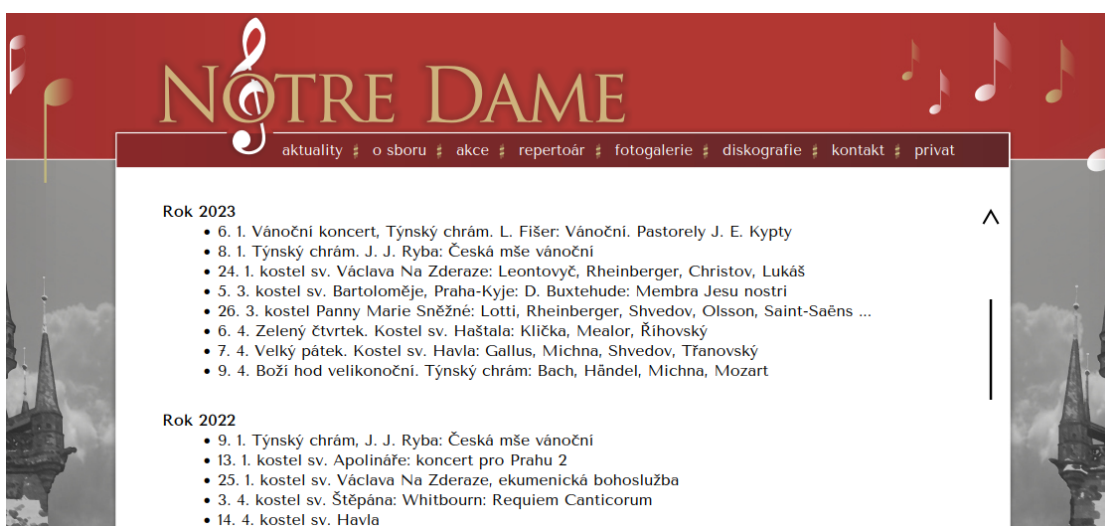
Aby prezentační i soukromá část plnily své poslání, je potřeba snadné přidávání obsahu. Ideálně aby se člověk nemusel zdlouhavě zdržovat formátováním a aby se mu přidáním jedné fotky nerozsypan celý zbytek. Dle rozhovoru se současnou webeditorkou to ale současné řešení neumožňuje, obsah je zarovnáván ručně, nikoliv automaticky, takže při změně obsahu (např. přidání nové aktuality či odebrání zastaralých informací) se často stane, že je potřeba přeformátovat půlku stránky, aby to jakž takž vypadalo a stránka nebyla úplně rozhozená. Při bližším zkoumání stránek si můžeme všimnout, že část formátování je jednoduše řešena novým řádkem, dokonce najdeme i „návod“, jak s daným „formátováním“ pracovat. (obr. 2.2) Co se týká designu dalších stránek, pro formátování byl například použit Microsoft Word, který byl pak nahrán na web (pomocí Save As možnosti). Toto řešení není příliš vhodné například proto, že když se něco na stránce náhodou pokazí, najít ve změní kódu věc, která je špatně, je téměř nemožné. Word totiž přidává spoustu vlastních tagů a informací, které nejsou ani potřebné, a soubor je pak zbytečně nepřehledný. Microsoft uvádí, že při tomto postupu může výsledek ztratit některé vlastnosti, mezi které patří například formátování znaků a odstavců, rozložení stránky či formátování grafiky a tabulek.[7]

²<https://www.google.com/intl/cs/sheets/about/>





■ Obrázek 2.3 Hlavní stránka webu Notre Dame, zdroj: [8]



■ Obrázek 2.4 Posuvník na okraji stránky a na kraji boxu s obsahem, zdroj: [8]

2.2.2 Pueri Gaudentes⁵

Webové stránky tohoto sboru působí na první pohled celkem hravě (obr. 2.5). Při přístupu z mobilního telefonu se sice stránka zbaví přebytečných okrajů, ale jiné přizpůsobení se nekoná, což příliš prohlížení neusnadňuje. Pro dostání se k dalším informacím o sboru, jako například k jednotlivým oddělením, vyžaduje kliknutí na „Sbor“ a pak se následně objeví další menu. Každý na toto řešení může mít jiný názor, nicméně autor této práce to nepovažuje za příliš šťastné řešení, protože to zbytečně ztěžuje orientaci na stránce. Na druhou stranu stránka obsahuje navigaci i v zápatí, což je velmi užitečné, především když se menu neposouvá spolu s obsahem, a tedy není vždy vidět, když uživatel dojde na konec obsahu.

Dalším prvkem k zamyšlení je způsob, jakým je realizováno menu. Přestože sbor není povinným

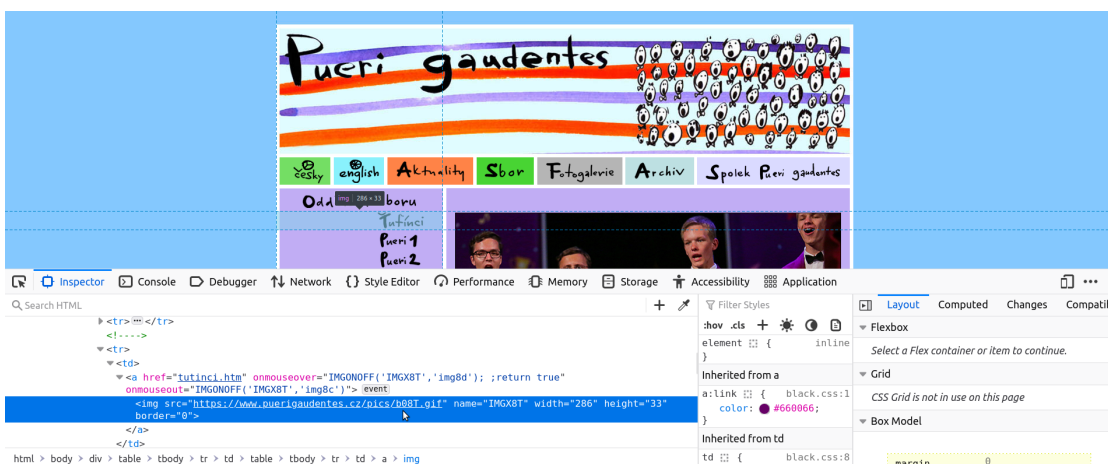
⁵<https://www.puerigaudentes.cz/uvod.htm> ([10])

subjektem, který má za povinnost mít přístupné webové stránky (viz např. [11]), většinu zřejmě napadne, že mít web alespoň trochu přívětivý pro osoby se zhoršeným zrakem, je dobrý nápad, už jen proto, že není neobvyklé, aby se mezi rodiči nacházel např. nevidomý, který potřebuje web pro přístup k informacím. Současné řešení ovšem používá místo textu v menu obrázky, bez jakéhokoliv návodného jména, chybí dokonce i alt (obr. 2.6). Podle [12] by přístupný web měl obsahovat ke každému netextovému obsahu textovou variantu, což zde splněno není a ztěžuje to tak porozumění webu zrakově postiženým.

Co se nastavení jazyka týká, na webových stránkách sice existuje možnost přepnutí do anglického jazyka, ale opět pouze pomocí jiných odkazů, a opět se stane, že při kliknutí na některou možnost (např. „Photo Gallery“) se stránka (menu) přepne opět do češtiny. V anglické verzi také úplně zmizí možnost dostat se na podrobnější informace ohledně sboru (jako jsou informace ohledně přípravných oddělení či plánovaných vystoupeních).



■ Obrázek 2.5 Hlavní stránka sboru Pueri Gaudentes, zdroj: [10]



■ Obrázek 2.6 Menu jako obrázky na stránkách Pueri, zdroj: [10]

2.2.3 Pražská kantiléna⁶

Prezentační část webových stránek Pražské kantilény dýchá moderním dojmem (obr. 2.7). Stránky se přizpůsobí i při prohlížení z mobilního telefonu – z horního menu se stane menu hamburgerové, obrázky s pěveckými sekcemi se seřadí pod sebe (obr. 2.8). Obrázky také obsahují textový popis a pro text je skutečně použit text, nikoliv obrázek textu. Při vytváření stránek byl pravděpodobně použit WordPress. Použití tohoto nástroje má hned několik výhod, jako například celkem nízké náklady na vytvoření a provoz stránek, velká možnost přizpůsobení stránek vlastním potřebám a nepříliš složité použití i pro netechnicky zdatné uživatele. ([14])

Stránka však neobsahuje možnost změny jazyka.



■ **Obrázek 2.7** Zobrazení úvodní stránky Pražské kantilény na počítači, zdroj: [13]

⁶<https://prazskakantilena.cz/> ([13])



PRAŽSKÁ KANTILÉNA



Dívčí sbor

■ **Obrázek 2.8** Zobrazení úvodní stránky Pražské kantilény na mobilním telefonu, zdroj: [13]

Kapitola 3

Praktická část

V následující kapitole se autor práce zaměří na vymezení hlavních požadavků na aplikaci a jejich splnění v rámci webové aplikace.

3.1 Požadavky na aplikaci

Při analýze požadavků na budoucí aplikaci je důležité si správně stanovit cíle. Nároky na budoucí systém dělíme na požadavky funkční a nefunkční, přičemž porozumění jejich rozdílům výrazně pomáhá k úspěšnému dokončení projektu a splnění očekávání ([15]). Hlavním rozdílem mezi nimi je, že „funkční požadavky popisují, co by měl systém dělat, zatímco nefunkční požadavky popisují, jak systém funguje“ [16].

3.1.1 Funkční požadavky

Po rozhovoru s odpovědnými členy sboru autor této práce vytyčil následující funkční požadavky:

Rozdělení veřejné a neveřejné části

Nejspíš nejdůležitějším požadavkem na novou webovou aplikaci je rozdělení stránek na veřejnou a neveřejnou část. Veřejná část slouží k prezentaci samotného tělesa veřejnosti. Součástí je prostor k představení sboru jako takového, jeho sbormistrů a dalších důležitých osob (klavíristka, hlasový korepetitor...), i např. vydaných CD. Důležitou součástí prezentace veřejnosti je také seznam koncertů a dalších událostí.

Neveřejná část naopak slouží k vystavení informací, které mají být veřejnosti skryté, avšak pro hladké fungování sboru jsou potřebné, jako například informace o čase srazu před koncertem, jaké noty vzít na zkoušku či jiná sdělení sbormistrů.

Jazykové mutace

Dalším funkčním požadavkem na web jsou jazykové mutace. Sbor se účastní tuzemských i zahraničních soutěží a festivalů a pro takové účely je dobré, aby si mohli i účastníci či porotci z jiných zemí něco o sboru přečíst. Překlad do anglického jazyka je v dnešní době základ, ale požadavek cílí i na nekomplikovanou možnost přidání dalších jazyků v budoucnu. Nastavení jazyka by mělo nějakou dobu vydržet, aby uživatel nestrávil půlku času neustálým přepínáním stránek do svého preferovaného jazyka.

Vzhledové sjednocení stránek

Aktuální stránky v době psaní této práce nemají jednotné formátování, ne u všech stránek se podařilo stejné nastavení, a tak některé stránky na pohled „odskakují“. Jeden z požadavků tedy cílí na toto. S tímto požadavkem také souvisí požadavek na možnost snadno prohlížet stránky z mobilu, jelikož v dnešní době již většina uživatelů prohlíží webové stránky ze svého mobilního telefonu.

Automatičnost a intuitivnost

Jedním z dalších funkčních požadavků je důraz na intuitivnost a co nejvíce automatických procesů. Velkou část členů sboru tvoří starší generace, pro kterou je často práce s počítačem a změna systému velký boj. Část práce také ušetří některé automatické činnosti, jako je například propisování jedné informace na více míst (místo nutnosti danou informaci psát znova – např. pro veřejnost je potřeba uvést čas začátku akce do sekce koncertů, ale členové sboru tuto informaci také potřebují, avšak do sekce koncertů obvykle nekoukají, protože potřebují také informaci o čase srazu, takže je pro ně přívětivější mít obě informace na stejném místě).

Přihlašování na akce

Důležitým funkčním požadavkem je také možnost zaznamenávání plánované účasti na nadcházejících akcích. Pro úspěšné zvládnutí koncertního vystoupení je nutné mít dostatek zpěváků. Ale jelikož ne každý má vždy čas, pro sbormistra je klíčové mít přehled o tom, kdo se kdy zúčastní, ideálně i z jakého je hlasu. Vznikl tak požadavek na snadné zaznamenávání (ne)účasti na akcích tak, aby každý člen mohl vyplnit informaci jen za sebe a omylem nezměnil údaje o někom jiném.

Více práv určité skupině členů

Přestože aplikace cílí na snadnou použitelnost a dostupnost pro všechny, i tak se může stát, že někdo není z různých důvodů schopný údaje samostatně vyplnit, a je třeba, aby je za něj vyplnil někdo jiný. Tím vznikl další požadavek, a to rozšířené oprávnění pro vybrané členy, které např. kromě možnosti vyplňování informací za jiné členy umožní i přidávání obsahu do neveřejné části, jako např. události, na které je třeba se přihlásit, či plánované skladby.

Adresář s kontakty

K výše zmíněnému požadavku se částečně vztahuje další funkční požadavek, a to prostor na uložení kontaktních informací členů. Jedná se o osobní údaje, a proto by je mělo vidět co nejméně lidí. Požadavkem je tedy skrytí kontaktních údajů běžným členům, avšak snadná dostupnost pro pověřené členy.

Sdělování informací pro členy

Speciálním požadavkem byl prostor pro sdělování sbormistra členům. Tento koncept funguje v současném řešení a požadavkem je toto zachovat. Sbornistr (či další oprávněné osoby) by měl mít možnost publikovat sdělení s libovolným obsahem (například odkaz na video z koncertu) ostatním členům, tedy zabezpečené proti očím veřejnosti.

3.1.2 Nefunkční požadavky

Nefunkční požadavky se zaměřují na vlastnosti systému, patří mezi ně například požadavky na:

- rychlost,

- dostupnost (kdy je systém dostupný – například jestli celý den, nebo jen v pracovní dobu),
- kapacitu (jaké jsou limity a co je systém schopný zvládnout),
- spolehlivost,
- technologii,
- vzhled (GUI),
- uživatelskou přívětivost (jak složité je systém používat pro koncového zákazníka).

([15]) Vzhledem k nevelké zkušenosti zadavatelů požadavků se systémy autor práce nedostal mnoho nefunkčních požadavků a řešení bylo ponecháno na něm.

3.2 Zvolení systémů

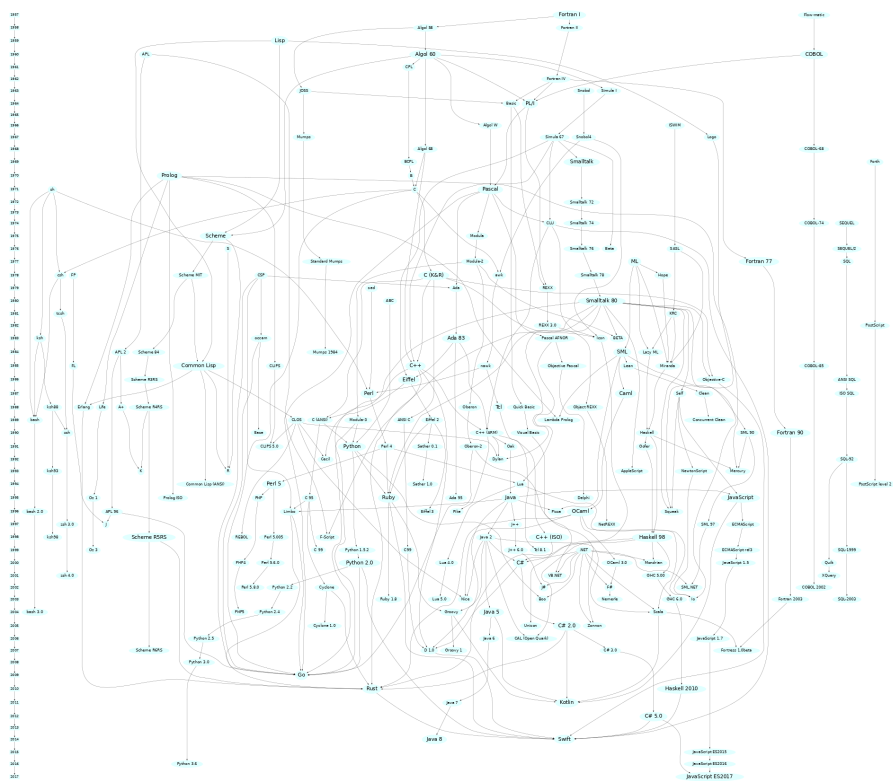
Před začátkem každého projektu si musí vývojář rozmyslet a rozhodnout, jaké systémy bude ve svém projektu používat. V dnešní době je výběr obrovský. Máme možnost si zvolit z několika programovacích jazyků, přičemž většina z nich nabízí možnost využití několika frameworků, dále, pokud potřebujeme ukládat data, si můžeme vybrat z několika databází (a to ještě nejméně ze dvou rozdílných typů databází) a několika možnostmi jak projekt zobrazovat.

Programovací jazyk

Jak už bylo zmíněno výše, programovacích jazyků existuje mnoho (obr. 3.1). Ne všechny se ale hodí pro všechno, proto je dobré se nad výběrem programovacího jazyka zamyslet a podívat se alespoň na některé výhody a nevýhody, které nabízí.

„Programovací jazyk je způsob, jakým programátoři (vývojáři) komunikují s počítači. Programovací jazyky se skládají ze souboru pravidel, která umožňují převádět řetězcové hodnoty na různé způsoby generování strojového kódu nebo, v případě vizuálních programovacích jazyků, grafických prvků. Obecně lze říci, že program je soubor instrukcí napsaných v určitém jazyce (C, C++, Java, Python) k dosažení určitého úkolu.“ ([17], přeloženo pomocí DeepL¹) Programovací jazyky mají také mnoho způsobů klasifikace, například na logické a imperativní, či na front-end a back-end jazyky. Front-endové jazyky se také nazývají client-side. Slouží k vytvoření virtuálních a interaktivních webových prvků, které uživatel může vidět. Mezi takové patří například HTML, CSS a JavaScript. Back-end jazyky, známé také jako server-side jazyky naopak vytváří logiku a funkčnost webové stránky, jako např. zpracování, uložení a spravování uživatelských účtů. Mezi takovéto jazyky patří např. Python, Java, či PHP. Z toho lze vidět, že pro zpracování většího projektu je mnohdy potřeba použít více programovacích jazyků.

¹<https://www.deepl.com/cs/translator>



■ **Obrázek 3.1** Diagram zobrazující historii a množství programovacích jazyků, zdroj: <http://rigaux.org/language-study/diagram.png>

PHP Jako programovací jazyk zvolil autor této práce jazyk PHP, a to především díky využitelnosti frameworků a předchozí zkušenosti z práce na školním projektu.

Framework Využití frameworku při vývoji webové aplikace má hned několik výhod. Mezi ty největší patří například úspora času vývoje, protože o část věcí se framework stará sám. Další výhodou je např. učení se dobrým programovacím návykům, jelikož mnoho frameworků využívá MVC přístup. Nepochybnou výhodou je také jistá míra zabezpečení, které je v dnešní době velmi důležité. Většina frameworků umí odvrátit cross-site scripting útok, či SQL injection. ([18])

MVC je jeden z typů třívrstvé architektury, která se v webovém vývoji používá. Jedná se o oddělení dat od logiky a od prezentace uživateli, přičemž každá vrstva může komunikovat pouze s vrstvou o jedna výše či níže. Modelová vrstva se stará o přístup k datům a databázi a komunikuje s controllerem, což je jakýsi prostředník, který ovládá většinu logických operací v aplikaci. Controller pak předává výstup viewu, který pak např. vykreslí webovou stránku a zobrazí na ní data. ([19])

3.3 Implementace aplikace

Při implementaci aplikace byly především využívány oficiální dokumentace systémů Symfony ([20]), Twig ([21]), Bulma ([22]). Užitečné poznatky byly také nalezeny v knižních publikacích ([23], [24], [25])

3.3.1 Zvolené systémy

Jako prostředek k naimplmentování aplikace zvolil autor této práce framework Symfony (s programovacím jazykem PHP), templatovací jazyk Twig, CSS framework Bulma a databázi PostgreSQL. Důvodem byly především zkušenosti z předchozí práce na školním projektu a vysoká oblíbenost těchto systémů, poskytující tak dobrou vývojářskou komunitu. Jedná se také o systémy s velmi dobrou dokumentací, která je při učení se s novými systémy, ale i při pozdější práci, velmi důležitá.

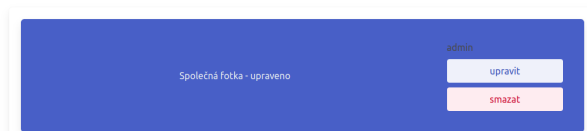
3.4 Uživatelské testování

V rámci uživatelského testování zkoušeli různí zástupci populace použít prototyp aplikace. Užitečnou zpětnou vazbu autor práce použil k opravení menších nedostatků a k naplánování vylepšení aplikace v dalších verzích.

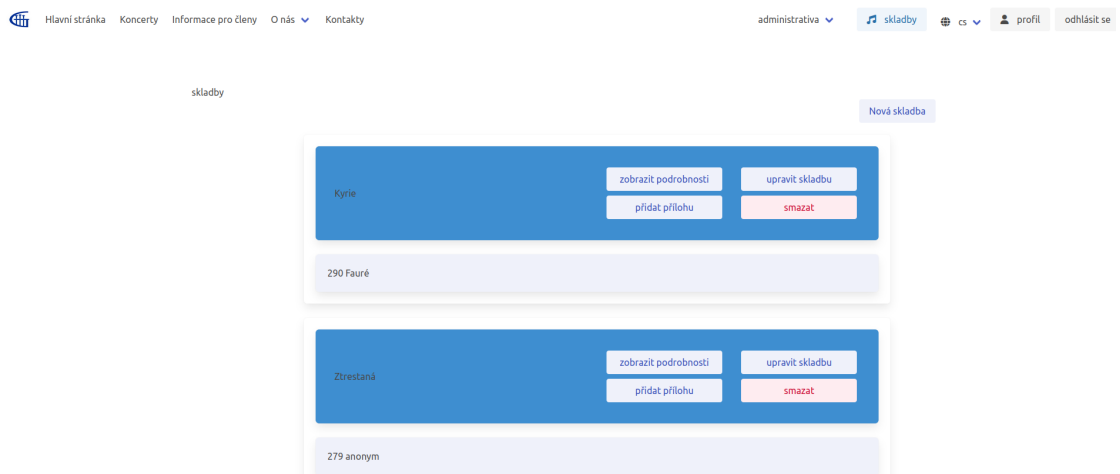
3.5 Ukázka aplikace



Obrázek 3.2 Hlavní stránka aplikace



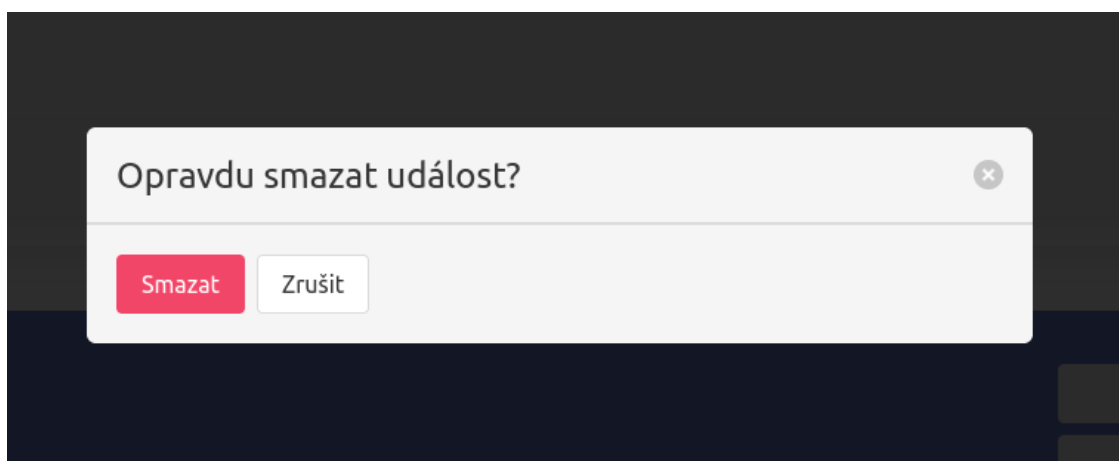
Obrázek 3.3 Zobrazení zpráv na neveřejné stránce



■ **Obrázek 3.4** Zobrazení skladeb



■ **Obrázek 3.5** Zobrazení rozhraní vedení pro správu členů



■ **Obrázek 3.6** Potvrzovací okénko před smazáním s možností návratu

■ Výpis kódu 3.1 Ukázka šablony Twig pro zobrazení příspěvků

```
{% for article in articles %}
  <div class="column is-three-quarters is-offset-2">
    <div class="box">
      <div class="box has-background-link">
        <div class="is-flex is-flex-direction-row
          is-align-items-center justify-content-center">
          <div class="column is-three-quarters has-text-white">
            {{ article.getTitle() | raw }}
          </div>
          <div class="column is-one-quarter">
            <div class="is-flex is-flex-direction-column
              is-justify-content-space-between">
              <span class="is-pulled-right m-1">
                {{ article.getAuthor().getFullName()}}
              </span>
              <br>
              {% if is_granted('ROLE_CAN_EDIT') %}
                {% if article.getAuthor() is same as (user) %}
                  <button class="button is-link is-light
                    is-pulled-right m-1" id="edit-article"
                    type="button" onclick="location.href=
                      '{{ path('editArticle',
                        {'id': article.getId}) }}'">upravit</button>
                {% endif %}
                <button class="js-modal-trigger button
                  is-danger is-light m-1"
                  data-target="modal-js-remove">
                  smazat</button>
              {% endif %}
            </div>
          </div>
        </div>
      </div>
      <div class="box has-background-link-light">
        {{ article.getContent() | raw }}
      </div>
    </div>
  </div>
  <div id="modal-js-remove" class="modal">
    <div class="modal-background"></div>
    <div class="modal-card">
      <header class="modal-card-head">
        <p class="modal-card-title">Opravdu smazat prispevek?</p>
        <button class="delete" aria-label="close"></button>
      </header>
      <footer class="modal-card-foot">
        <button class="button is-danger" onclick="location.href=
          '{{ path('deleteArticle', {'id': article.getId}) }}'">
          Smazat</button>
        <button class="button">šZruit</button>
      </footer>
    </div>
  </div>
{% endfor %}
```


Zhodnocení splnění cílů a možnosti dalšího rozšíření.

4.1 Splnění cílů

Hlavní cíle bakalářské práce byly splněny. Autor práce se zaměřil na hlavní nedostatky současného řešení a řešení webových stránek jiných sborů a zprovoznil prototyp aplikace. Cíl, který nebyl úplně splněn, je samotné nasazení webové aplikace, a to především z důvodu finálního rozhodnutí o hostování aplikace, které je zatím stále v řešení pověřenými členy sboru a autor tohoto textu to příliš neovlivní.

4.2 Možnosti rozšíření

Aplikace rozhodně není dokonalá a je zde prostor k vylepšením. Mezi hlavní inovace plánované v příštích verzích patří lepší přizpůsobení vzhledu na menších displejích a přidání dalších funkcí, jako například fotogalerie či přidání přílohy ke skladbě přímo při prvním zaznamenávání údajů o skladbě. Nabízí se také prostor pro upravení zobrazení seznamu skladeb, například přidáním voleb na různé způsoby řazení skladeb (abecedně, vzestupně či sestupně podle čísla, podle autora...). Jako další rozšíření se také nabízí dokončení jazykové mutace pro anglický jazyk a přidání dalších jazyků.

V budoucnu by aplikace mohla obsahovat notifikace na nové příspěvky, nevyplněné účasti či nastavení zaslání upomínek a novinek na e-mail.

Návod na spuštění a používání aplikace

A.1 Návod na spuštění aplikace

1. Stáhněte a nainstalujte si Docker¹ a spusťte ho.
2. Otevřete příkazovou řádku v adresáři projektu (adresář *app*).
3. Proveďte příkazy `sudo docker-compose build --pull` a `sudo docker-compose up -d`, které rozběhnou dockerovský kontejner.
4. Nahrajte data do databáze: příkazem `sudo docker exec -it gaudium-db /bin/bash` se dostanete dovnitř kontejneru a pomocí `psql -U pguser -d public < /data/data.sql` načlňte databázi testovacími daty.
5. Otevřete webový prohlížeč a na localhostu² naleznete aplikaci.

A.2 Návod k použití

Při vývoji a implementaci aplikace byl kladen důraz na intuitivnost a jednoduchost, popisky v aplikaci by tedy měly být samovysvětlující a návodné.

Uživatel se musí nejprve zaregistrovat. Po registraci se již může uživatel přihlašovat do aplikace pomocí tlačítka *přihlásit se*. Po registraci je uživatel pouze obecným uživatelem, další možnosti jsou přístupné až po udělení vyšší role – např. člen. Role může přidávat administrátor, sbormistr či uživatel s rolí výbor.

Člen pak může například:

- zobrazit podrobnější informace k plánovaným akcím (jako např. čas srazu),
- přihlásit se na událost,
- zobrazit si přehled skladeb a jejich případných příloh (např. noty, MIDI nahrávka),
- zobrazit si soukromá sdělení od vedení sboru členům,
- vyplnit svůj profil s kontaktními údaji.

¹<https://www.docker.com/products/docker-desktop/>

²<http://localhost:80/>

Bibliografie

1. *Gaudium Praha* [online] [cit. 2023-01-20]. Dostupné z: <http://gaudiumpraha.org/>.
2. *Responzivní web: jak vypadá a proč ho mít* [online]. Active24 [cit. 2023-02-26]. Dostupné z: <https://www.active24.cz/jak-na-tvorbu-webu/tvorba-stranek-pokrocila/responzivni-web-jak-vypada-a-proc-ho-mit>.
3. KUBÍK, Milan. *Co znamená responzivní web a proč ho mít* [online]. 2021 [cit. 2023-02-26]. Dostupné z: <https://www.webnia.cz/deje-se/co-znamená-responzivni-web-a-proc-ho-mit>.
4. MARCOTTE, Ethan. *Responsive Web Design* [online]. 2010 [cit. 2023-02-26]. Dostupné z: <https://alistapart.com/article/responsive-web-design/>.
5. O'HARE, Sean. *Reasons Why a Multi-Language Site Benefits Your Business* [online]. 2023 [cit. 2023-04-20]. Dostupné z: <https://weglot.com/blog/reasons-why-a-multi-language-site-benefits-your-business/>.
6. (CISO), Chief Information Security Officer. *Shared Accounts* [online]. University of Washington [cit. 2023-01-20]. Dostupné z: <https://ciso.uw.edu/education/risk-advisories/shared-accounts/>.
7. *Limitations when you save a Word document as a web page* [online]. Microsoft Corporation [cit. 2023-01-20]. Dostupné z: <https://support.microsoft.com/en-us/topic/limitations-when-you-save-a-word-document-as-a-web-page-f361de08-ca4c-bc53-11ef-138c0e405c44>.
8. *Sbor Notre Dame* [online]. 2012 [cit. 2023-01-20]. Dostupné z: <http://www.sbornotredame.cz/>.
9. PULKITAGARWAL03PULKIT. *Difference between JSP and HTML* [online]. Geeksfor-Geeks [cit. 2023-01-20]. Dostupné z: <https://www.geeksforgeeks.org/difference-between-jsp-and-html/>.
10. *Pueri Gaudentes* [online] [cit. 2023-01-20]. Dostupné z: <https://www.puerigaudentes.cz/uvod.htm>.
11. *Zákon č. 99/2019 Sb., o zpracování osobních údajů a o změně souvisejících zákonů* [online]. Zakonyprolidi.cz, 2019 [cit. 2023-01-20]. Dostupné z: <https://www.zakonyprolidi.cz/cs/2019-99>.
12. PAVLÍČEK, Radek. *Web Content Accessibility Guidelines (WCAG): seznamte se, prosím* [online]. Poslepu.cz, 2019-02 [cit. 2023-01-20]. Dostupné z: <https://poslepu.cz/web-content-accessibility-guidelines-wcag-seznamte-se-prosim/>.
13. *Pražská kantilena* [online]. 2023 [cit. 2023-01-20]. Dostupné z: <https://prazskakantilena.cz/>.

14. NEWCOMER, Colin. WordPress Review. *ThemeIsle Blog* [online]. 2023 [cit. 2023-01-20]. Dostupné z: <https://themeisle.com/blog/wordpress-review/#gref>.
15. GORBACHENKO, Pavel. *Functional Requirements vs. Non-Functional Requirements: What's the Difference?* [Online]. Enkonix [cit. 2023-01-24]. Dostupné z: <https://enkonix.com/blog/functional-requirements-vs-non-functional/>.
16. MORRISON, Roger. *Rozdíl mezi funkčními a nefunkčními požadavky* [online]. strephonsays [cit. 2023-01-24]. Dostupné z: <https://cs.strephonsays.com/functional-and-non-functional-requirements-3325>.
17. VEERARAGHAVAN, Sruthi. *20 Most Popular Programming Languages to Learn in 2023* [online]. 2023 [cit. 2023-02-22]. Dostupné z: <https://www.simplilearn.com/best-programming-languages-start-learning-today-article>.
18. M., Will. *Best PHP Frameworks in 2023: Which One Should You Choose?* [Online]. 2023 [cit. 2023-02-22]. Dostupné z: <https://www.hostinger.in/tutorials/best-php-framework>.
19. SVIRCA, Zanfina. *Everything You Need to Know About MVC Architecture* [online]. 2020 [cit. 2023-02-22]. Dostupné z: <https://towardsdatascience.com/everything-you-need-to-know-about-mvc-architecture-3c827930b4c1>.
20. *Symfony*. Dostupné také z: <https://symfony.com/>.
21. *Twig – The flexible, fast, and secure template engine for PHP*. Dostupné také z: <https://twig.symfony.com/>.
22. *Bulma: A Modern CSS Framework*. Dostupné také z: <https://bulma.io/>.
23. SUEHRING, Steve; VALADE, Janet. *PHP, MySQL, JavaScript & HTML5 all-in-one for dummies*. Somerset: Wiley, 2013. For dummies. ISBN 9781118213704.
24. POREBSKI, Bartosz. *Building applications with Symfony, CakePHP, and Zend Frameworks*. 1st edition. Indianapolis, IN: Wiley, 2011. Wrox programmer to programmer Building PHP applications with Symfony, CakePHP, and Zend Framework. ISBN 1-283-37455-2.
25. ARMAND, Sébastien. *Extending symfony2 web application framework : optimize, audit, and customize web applications with symfony*. Birmingham, England: Packt Publishing Ltd, 2014. Community experience distilled. ISBN 1-78328-720-9.

Obsah přiloženého zip souboru

README.md	stručný popis obsahu adresáře
license.txt	informace o licenci
vlkova.pdf	text práce ve formátu PDF
app	adresář se soubory aplikace
bin		
config	adresář s konfiguračními soubory
data	adresář s připravenými testovacími daty pro nahrání do databáze
migrations	adresář s databázovými migracemi
public		
style	adresář s CSS soubory
favicon.ico	ikonka zobrazená v prohlížeči
index.php	vstupní bod aplikace
src	adresář se zdrojovými kódy
templates	adresář s Twig šablonami stránek
translations	adresář se soubory používanými pro jazykové přizpůsobení
Caddyfile	konfigurační soubor pro Caddy server
composer.json	soubor s definicemi závislostí a konfigurací pro Composer
composer.lock	..	soubor obsahující seznam přesných verzí nainstalovaných závislostí pro Composer
docker-compose.yml	soubor pro konfiguraci a orchestraci Docker kontejnerů
Dockerfile	soubor pro definici Docker image (obrazu)
symfony.lock	soubor obsahující seznam verzí nainstalovaných balíčků pro Symfony
.env	soubor s prostředím pro běh aplikace