

Bakalářská práce



F3 Fakulta elektrotechnická
Katedra počítačů

Webová aplikace pro podporu výtvarného ateliéru

Daria Kulynychenko

Vedoucí: Ing. Pavel Náplava, Ph.D.
Studijní program: Softwarové inženýrství a technologie
19. května 2019

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Kulynychenko** Jméno: **Daria** Osobní číslo: **465896**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávací katedra/ústav: **Katedra počítačů**
Studijní program: **Softwarové inženýrství a technologie**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Webová aplikace pro podporu výtvarného ateliéru

Název bakalářské práce anglicky:

Web appliaction for creative studio support

Pokyny pro vypracování:

Navrhněte a implementujte webovou aplikaci, která umožní výtvarnému ateliéru, nabízející zájemcům kurzy kreslení „pravou“ mozkovou hemisférou. Smyslem aplikace je vytvořit komunikační kanál mezi ateliérem a účastníky kurzů, v rámci kterého se budou sdílet návody pro kreslení v podobě foto a video návodů. Postupujte následujícím způsobem:

- 1) Identifikujte a analyzujte požadavky výtvarného ateliéru.
- 2) Na základě požadavků vyberte vhodnou platformu pro realizaci aplikace a vytvořte návrh aplikace.
- 3) Aplikaci implementujte a ověřte její funkčnost vůči požadavkům zadavatele.
- 4) Společně se zadavatelem aplikaci nasadte do reálného prostředí a uživatelsky ověřte, že splňuje požadavky zadavatele.
- 5) Popište způsob používání aplikace a jejího provozu.

Seznam doporučené literatury:

- [1] UML 2 Use Case Diagramming Guidelines [online]. Dostupné z: <http://www.agilemodeling.com/style/useCaseDiagram.htm>.
- [2] FOWLER, M. Destilované UML. Knihovna programátora. Grada, 3 edition, 2009. ISBN 9788024720623.
- [3] SILVER, Bruce (Bruce Richard). BPMN method and style: with BPMN implementer's guide. Aptos, Calif. : Cody-Cassidy Press, 2011. ISBN 9780982368114
- [4] BPMN Specification - Business Process Model and Notation [online]. Dostupné z: <http://www.bpmn.org/>.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Pavel Náplava, Ph.D., katedra ekonomiky, manažerství a humanitních věd FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **13.02.2019**

Termín odevzdání bakalářské práce: **24.05.2019**

Platnost zadání bakalářské práce: **20.09.2020**

Ing. Pavel Náplava, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Ing. Pavel Ripka, CSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Studentka bere na vědomí, že je povinna vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

_____ Datum převzetí zadání

_____ Podpis studentky

Poděkování

Především bych chtěla poděkovat vedoucímu mé práce Ing. Pavlu Náplavovi, Ph.D., za jeho cenné rady, věcné připomínky a vstřícnost při konzultacích a vypracování této práce. Také bych chtěla poděkovat Pavle Macháčkové a ateliéru Mozaika za spolupráci a čas věnovaný tomuto projektu.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracovala samostatně a že jsem uvedla veškerou použitou literaturu.

V Praze, 19. května 2019

Abstrakt

Cílem této práce bylo provést analýzu, vytvořit návrh a implementovat webovou aplikaci pro výtvarný ateliér. Aplikace slouží jako komunikační kanál mezi ateliérem a jeho zákazníky. Jsou do ní pravidelně vkládány nové návody, jak kreslit a malovat, a to v podobě foto- a videonávodů. Kromě analýzy požadavků a samotného vývoje aplikace bylo provedeno porovnání a výběr vhodného nástroje pro tuto činnost, uživatelské testování a nasazení do reálného prostředí.

Klíčová slova: ateliér, webová aplikace, CMS, analýza, WordPress

Vedoucí: Ing. Pavel Náplava, Ph.D.

Abstract

The aim of this work was to analyze, create a proposal and implement a web application for the art studio. The application serves as a communication channel between the studio and customers. The studio regularly inserts new tutorials on how to draw and paint in the form of photo and video tutorials. In addition to analyzing requirements and development, a comparison and selection of a suitable tool for application development, user testing and deployment in the real environment was performed.

Keywords: art studio, web application, CMS, analysis, WordPress

Title translation: Web application for creative studio support

Obsah

1 Úvod	1	4.3 WordPress	19
2 Seznámení se s problémem	3	4.4 Drupal	20
2.1 Představení zákazníka	3	4.5 Joomla!	21
2.2 Sběr požadavků	4	4.6 Ghost	22
2.3 Návrh řešení identifikovaného problému	5	4.7 Porovnání CMS systémů	23
3 Specifikace požadavků zákazníka	7	4.8 Výsledek porovnání	26
3.1 Katalog požadavků	7	5 WordPress	27
3.2 Diagram případů užití	9	5.1 Technologie a instalace	27
3.3 Domain model	10	5.2 Plocha administrátora	29
3.4 Process model	11	5.3 Editor	30
3.5 Návrhy obrazovek	12	5.4 Pluginy a vzhledy	31
3.6 Shrnutí kapitoly	14	5.5 Aktualizace a kompatibilita	31
4 CMS systémy	17	5.6 Základní prvky - uživatelé, role, příspěvky, taxonomie	32
4.1 Definice CMS systémů	18	5.6.1 Uživatelé a jejich role	32
4.2 Výběr CMS systému pro vývoj aplikace Joaha	18	5.6.2 Příspěvky a vlastní typy	33
		5.6.3 Taxonomie	34

6 Průběh implementace aplikace	35	7.2.4 Scénář U4	52
6.1 Vývojové prostředí	36	7.2.5 Scénář U5	53
6.2 Průběh implementace požadavků	36	7.3 Výsledky testování	53
6.3 Použité pluginy	38	8 Provoz aplikace	55
6.3.1 Online kurzy	39	9 Vyhodnocení práce z pohledu projektu	57
6.3.2 Členství	41	9.1 Rizika	58
6.3.3 Vzhled a rozvržení, šablony . .	43	9.2 Řízení změn	58
6.4 Vlastní úpravy kódu	45	9.2.1 Změna požadavků	58
6.5 Vzhled aplikace	46	9.2.2 Změna výstupu projektu	59
6.6 Shrnutí procesu implementace . .	47	10 Aktuální stav práce	63
7 Testování	49	11 Vyhodnocení úspěšnosti projektu	65
7.1 Testovací scénáře – administrátorská část	50	12 Závěr	67
7.1.1 Scénář A1	50	Literatura	69
7.2 Testovací scénáře – uživatelská část	51	Seznam zkratk	75
7.2.1 Scénář U1	51	Slovník pojmů	77
7.2.2 Scénář U2	51	A Specifikace	79
7.2.3 Scénář U3	52		

A.1 Požadovaná funkcionality	79
A.2 Komentované návrhy obrazovek	81
B Katalog požadavků	83
C Use case diagram	85
C.1 Případy užití pro administrátora	85
C.2 Případy užití pro nepřihlášeného uživatele	87
D Domain model - online kurz	89
E Harmonogram	91
F Pracovní posudek	93
G Struktura CD	95

Obrázky

3.1 Diagram aktérů	10
3.2 Proces získání členství zdarma .	12
3.3 Placený videonávod pro uživatele bez členství	15
5.1 Popis instalace od WordPress ..	28
5.2 Plocha administrátora	30
6.1 Neplacený fotonávod – wireframe a design	47
6.2 Profil uživatele - wireframe a design	48

Tabulky

4.1 Porovnání CMS nástrojů	25
6.1 Porovnání pluginů - online kurzy	41
6.2 Porovnání pluginů - členství....	43
6.3 Porovnání pluginů – tvorba vzhledů.....	45
9.1 Rizika projektu.....	60
9.2 Katalog změnových požadavků .	61



Kapitola 1

Úvod

Cílem této práce je provést analýzu požadavků zákazníka a následně navrhnout a vyvinout aplikaci, která je uspokojí. Nedílnou součástí zadání je testování aplikace zákazníkem a její nasazení do reálného prostředí. Zákazník také obdrží kompletní dokumentaci a uživatelské manuály. Aplikace bude sloužit jako komunikační kanál mezi výtvarným ateliérem a jeho klienty a zároveň jako prostředek inspirace pro uživatele.

Zvolila jsem si toto téma z několika důvodů. Za prvé ve volném čase ráda kreslím a maluji, a proto taková práce je pro mě zajímavá. Za druhé mě zaujala praktická část, a to seznámení se s novou technologií. Nakonec považuji za přínosné těsnou spolupráci s reálným zákazníkem.

V následujících kapitolách detailněji popíšu zákazníka a proces sběru požadavků a také identifikuji problém, který je nutné řešit během naší spolupráce. Specifikuji požadavky a provedu jejich analýzu pomocí UML a BPMN diagramů. Také popíšu proces vytvoření a použití návrhů obrazovek, nazývaných také *Wireframes*, které byly nedílnou součástí analýzy a návrhu. V kapitole 4 porovnam nástroje, které jsou vhodné pro vývoj aplikace, a vyberu jeden z nich. V praktické části práce se zaměřím na základní prvky vybraného CMS systému a průběh implementace požadavků. Také uvedu metody testování a popis testovacích scénářů. Poslední kapitoly jsou věnovány vyhodnocení projektu a popisu jeho aktuálního stavu.

Kapitola 2

Seznámení se s problémem

Kapitola slouží úvodem do problému, řešení kterého je popsáno v této práci. V následujících podkapitolách představím zákazníka a popíšu proces sběru jeho požadavků. Jako shrnutí kapitoly navrhnu řešení problému, které bude implementováno během naší spolupráce.

2.1 Představení zákazníka

Zákazníkem je Ateliér Mozaika [1] – výtvarný ateliér, který nabízí kurzy kreslení a malby. Má velké množství klientů a jejich počet se stále zvyšuje, proto se před několika lety rozhodl připravit pro ně dárek. Majitelka ateliéru přišla s nápadem vytvořit aplikaci, která bude poskytovat možnost naučit se kreslit přímo z domova. Aplikace bude obsahovat foto- a videonávody. Uživatelé budou mezi sebou komunikovat, sdílet své zkušenosti a prostě se inspirovat. Benefitem pro stávající zákazníky bude bezplatný přístup do některých kategorií.

Po provedení výzkumu mínění mezi klienty ateliéru bylo zjištěno, že lidé preferují webovou aplikaci s responzivním designem, aby měli možnost pracovat i z mobilního zařízení. Hlavní výhodou webové aplikace je to, že je snadno přenositelná a dostupná z libovolného zařízení, které má přístup na internet.

Aplikace bude mít další funkčnosti, jako je přidání návodů uživatelem,

- Would not (zatím nebude mít) – požadavky, které nebudou po dohodě se zákazníkem realizované v této verzi, ale možná budou uspokojeny v budoucích verzích systému.

Po několika setkáních byla specifikace připravena a následně byla schválena zákazníci, čímž byl stanoven rozsah projektu. Příklad specifikace se nachází v příloze A, celou verzi je možné najít na přiloženém CD.

2.3 Návrh řešení identifikovaného problému

Jako shrnutí této části definuji řešení problému, které má být implementováno během spolupráce se zákazníkem.

Ateliér Mozaika potřebuje webovou aplikaci, která bude sloužit jako inspirace pro jeho klienty. Systém jim bude poskytovat přístup k foto- a videonávodům, online kurzům a obrázkům. Zobrazovaný obsah se má měnit v závislosti na roli uživatelů, kteří musí mít možnost komunikovat mezi sebou, a to pomocí komentářů a vlastních návodů. Funkcionalita aplikace „Joaha“ musí odpovídat specifikaci, popsané v podkapitole 2.2. Nedílnou součástí zadání je návrh aplikace takovým způsobem, aby byla snadno rozšiřitelná o další funkcionalitu, definovanou ve specifikaci s prioritou „**Would not**“.

Kapitola 3

Specifikace požadavků zákazníka

Pro analýzu požadavků a návrh řešení jsem se rozhodla použít UML a BPMN diagramy. Ve své práci jsem používala UML k tvorbě náčrtků¹, tedy pro lepší pochopení požadavků a usnadnění komunikace se zákazníkem. K zachycení interakce uživatele a systému jsem použila *diagram případů užití*, pro vytvoření slovníku pojmů *domain model*. BPMN jsem používala pro grafické znázornění procesů v aplikaci – vytvoření business process diagramu.

Při vytvoření diagramů jsem se řídila příklady, návody a radami z knih Martina Fowlere „Destilované UML“ [43] (UML diagramy), Craige Larmana „Applying UML and patterns“ [47] (domain model), Bruce Silvere „BPMN method and style“ [51] (návrh business process diagramu) a ze specifikace notace BPMN [3], dostupné z oficiálních stránek OMG, věnovaných této problematice².

3.1 Katalog požadavků

V kapitole 2.2 bylo zmíněno, že veškeré požadavky zákazníka byly zaznamenány do specifikace. Nebyly však ve formě vhodné pro další analýzu a vývoj. Specifikace popisovala obecný pohled na systém a jeho funkcionalitu, které bylo nutné blíže upřesnit. Proto jsem se rozhodla vytvořit katalog požadavků s jednoduchou strukturou:

¹Anglicky: UML as sketch

²Dostupné z <http://www.bpmn.org/>

1. ID požadavku – nutné pro jednoduché odkazování na požadavek.
2. Název – pro snadnou navigaci katalogem.
3. Detailní popis požadavku – stručné shrnutí toho, co má systém poskytovat nebo umožňovat uživateli.
4. Typ – funkční, nebo nefunkční³ požadavek.
5. Priorita – převzata ze specifikace, použit byl stejný systém MoSCoW.
6. Složitost – tento prvek je mým odhadem na základě představy o možných nástrojích, které budou použity pro vývoj. Jejich detailnější popis je uveden v kapitole 4. Pro definici stupně složitosti jsem používala klasický systém *High, Medium, Low*, což v českém překladě znamená vysoká, střední a malá složitost.
7. Vazby na jiné dokumenty – vazby na specifikaci (bod specifikace, z kterého byl požadavek odvozen), nebo případ užití (ID případu užití, který se vztahuje k tomuto požadavku).
8. Uživatelská role – role uživatele, pro kterého je tento požadavek relevantní, tzn. něco mu umožňuje nebo ho v něčem omezuje.
9. Stav požadavku – tato informace se bude hodit pro vývoj aplikace. Na začátku byly všechny požadavky ve stavu „ToDo“, což znamená, že byly ve frontě na zpracování.
10. Zdroj požadavku – hlavním zdrojem je zde samozřejmě zákazník. Existují ale požadavky, které plynou z legislativy (například, požadavek na ochranu osobních údajů). Proto si myslím, že je nutné tento údaj u požadavků zaznamenat.
11. Poznámky – sem se zaznamenávají definice pojmů, detailnější vysvětlení požadavku a další důležité informace.
12. Štítky – pro rychlé filtrování požadavků podle témat.

Jako prostředek pro vytvoření katalogu požadavků jsem použila MS Excel. Je to komerční nástroj k tvorbě tabulek. Pro mé účely byl vhodný zejména proto, že poskytuje jednoduchý způsob údržby, řazení a filtrace tabulek. Dalším důvodem výběru MS Excel byl výstupní formát *.xls*, který je dnes snadno přenositelný a může být otevřen v mnoha aplikacích. Náhled katalogu požadavků se nachází v příloze B, jeho celá verze je dostupná na příloženém CD.

Katalog požadavků v této práci má několik cílů. Za prvé došlo k upřesnění požadavků zákazníka a jejich rozdělení na menší části, za druhé bylo nutné

³anglický pojem je Non-functional

specifikovat požadavky, jejichž zdrojem je legislativa. Takové požadavky se považují za nezbytné při vývoji libovolného podobného systému. Důležité bylo také vytvoření komplexního pohledu na vyžadovanou funkcionalitu, který by zahrnoval nejen prioritu, ale i složitost a souvislost s jinými důležitými prvky analýzy.

3.2 Diagram případů užití

Jak jsem již zmínila dříve, při analýze požadavků jsem použila diagram případů užití⁴, zpracovaný v notaci UML. Diagram případů užití ukazuje vztahy mezi aktéry a případy užití v rámci systému[26]. Nejdůležitějším cílem diagramu je znázornit, jak může uživatel v určité roli používat systém. Ve své práci jsem používala případy užití hlavně k popisu omezení přístupu uživatelů do funkčnosti systému.

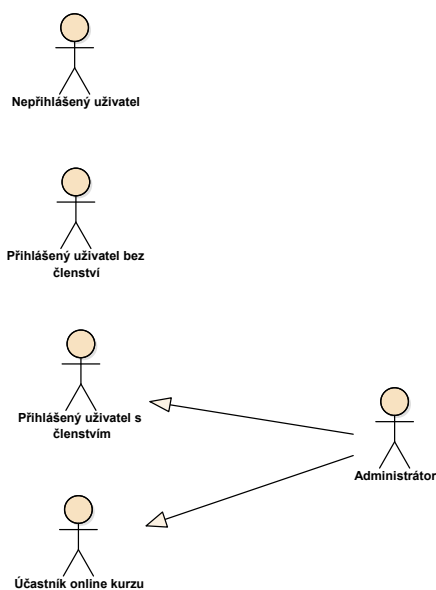
Aplikace „Joaha“ podporuje více rolí uživatelů, které mají svá omezení, proto je nutné jasně definovat, co který uživatel bude moci v aplikaci dělat. Z tohoto důvodu a za účelem přehlednosti jsem rozdělila velký diagram do několika menších. Každý diagram má právě jednoho aktéra, tj. uživatelskou roli, která je propojena s případy užití, jež jsou pro ni určeny. Takový způsob rozdělení diagramu má své výhody a nevýhody. **Výhodou** je hlavně to, že lze jednoduše zjistit, co je pro danou uživatelskou roli povoleno. Následná kontrola konzistence diagramu a požadavků zákazníka je pak snadnější. **Nevýhodou** je, že se v diagramech vyskytují duplicity jelikož velké množství případů užití je relevantní pro více rolí. Po zvážení všech pro a proti jsem se rozhodla, že přehlednost a možnost rychle zjistit omezení uživatelských rolí jsou pro mě důležitější než absence duplicit.

Jednou z důležitých součástí diagramu případů užití je diagram aktérů, který slouží k tomu, aby zdůraznil vztahy mezi uživatelskými rolemi, zvláště generalizaci⁵ aktéra.

Jak vidíme na obrázku 3.1, administrátor je specializací účastníka online kurzu a zároveň přihlášeného uživatele s členstvím. To znamená, že je mu poskytnuta všechna funkcionalita obou aktérů a ještě nějaká navíc. Proto v samotném diagramu případů užití jsou pro administrátora zobrazeny jen ty případy užití, které jsou určeny **jen** jemu (příloha C.1).

⁴Use Case diagram

⁵anglicky – generalization



Obrázek 3.1: Diagram aktérů

Diagram případů užití pro nepřihlášeného uživatele je uveden v příloze C.2, celý diagram včetně souboru aplikace Enterprise Architect je k nahlédnutí na přiloženém CD.

3.3 Domain model

Aplikace „Joaha“ pracuje s mnoha pojmy, které se vzájemně podobají, jako jsou „členství“ a „přístup“, „víkendový kurz“ a „online kurz“. Proto jsem se rozhodla pro jejich jednoznačnou definici použít doménový model. Na začátku jsem definovala všechny entity z reálného světa. Pro tento účel jsem použila metodu použití seznamu kategorií popsanou v knize „Applying UML and patterns“ Graige Larmana[47]. Tato metoda spočívá v tom, že se pro každou z kategorií z tohoto seznamu snažíme najít objekt, který by do ní patřil. Například, do kategorie „Transakce“ patří v případě aplikace „Joaha“ *Platba*, *Nákup*, do kategorie „Ostatní připojené systémy“ pak *Platební brána* a *Sociální síť*.

Objekty nalezené pomocí této metody jsem rozdělila do několika balíčků v závislosti na jejich kategorii:

- Role – například administrátor, přihlášený uživatel.
- Kontejnery – historie nákupů, „oblíbené“.
- Média – návody, online kurzy.
- Transakce – jak bylo zmíněno výše, patří sem nákup a platba.
- Externí systémy – sociální síť a platební brána.
- Ostatní – sem patří objekty jako členství, datum, komentář.

V systému takové velikosti, jako je aplikace „Joaha“, je doménový model poměrně komplexní a velký. Z tohoto důvodu ho nedávám celý do příloh. V příloze D je uvedena část diagramu, která se týká online kurzů. Celý diagram, stejně jako i veškerá dokumentace, je k nahlédnutí na přiloženém CD.

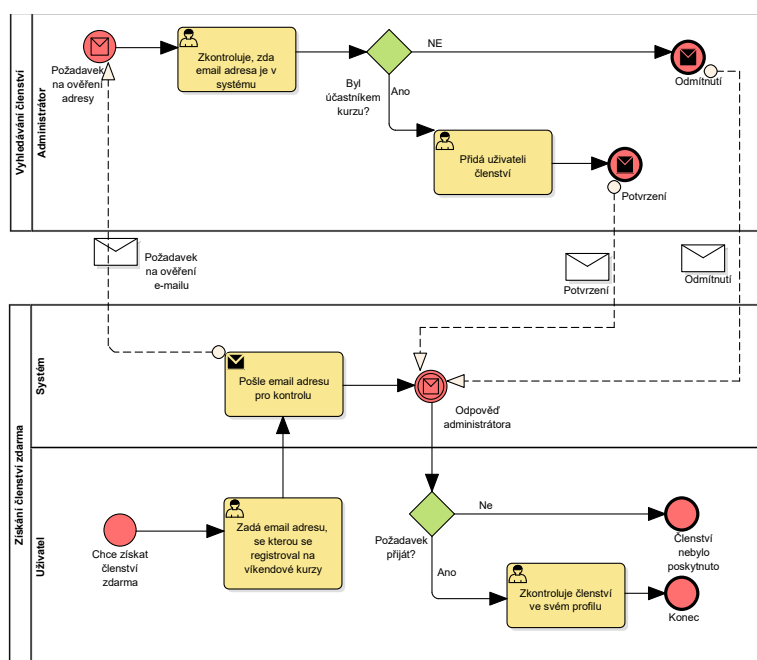
3.4 Process model

Protože aplikace „Joaha“ má komplexní funkcionalitu, rozhodla jsem se její nejsložitější procesy zachytit v modelu procesů. Pro tento účel jsem se rozhodla použít notaci BPMN 2.0, nikoli UML. Diagram vytvořený pomocí této notace považuji za čitelnější pro zákazníka, který nemá zkušenost v oblasti informačních technologií.

Velké množství procesů, které budou probíhat v aplikaci „Joaha“, je triviální, například komentování, přidání návodu nebo vyhledávání. Přesto obsahuje aplikace i procesy, které jsou za prvé velmi důležité z pohledu priorit požadavků, za druhé jsou netriviální a vstupují do nich i externí systémy. Proto jsem se rozhodla vizualizovat 4 procesy:

1. Přihlášení pomocí sociálních sítí.
2. Registrace pomocí sociálních sítí.
3. Získání členství zdarma z důvodu účasti ve víkendových kurzech.
4. Nákup členství.

V této podkapitole popíšu jeden z procesů – získání členství zdarma. Ostatní jsou dostupné na přiloženém CD.



Obrázek 3.2: Proces získání členství zdarma

Na obrázku 3.2 vidíme, že se celý proces získání členství zdarma začíná žádostí uživatele. Pokud se uživatel zúčastnil víkendových kurzů, pošle požadavek na poskytnutí členství. Administrátor musí ručně zkontrolovat, zda je e-mailová adresa v seznamu zákazníků, a na základě toho buď poskytnutí členství schválí, nebo nikoliv. Uživatel pak dostane notifikaci o vyřízení svého požadavku a následně své členství zkontroluje v osobním profilu.

V první verzi aplikace „Joaha“ probíhá kontrola z pohledu administrátora ručně. Důvodem je to, že zákazník v současné době mění systém evidence návštěvníků víkendových kurzů, proto ještě není definováno, který systém se napojí na aplikaci „Joaha“. V závislosti na termínu ukončení migrace dat do nového systému bude možné požadavek o automatické kontrole přidat jako požadavek na změnu⁶, nebo ho zahrnout do dalších verzí aplikace.

3.5 Návrhy obrazovek

Tato podkapitola je věnována návrhům obrazovek (wireframes) pro aplikaci „Joaha“. Ty slouží několika účelům [27]:

⁶anglicky – Change Request

1. Pomáhají zúčastněným stranám lépe porozumět tomu, co se bude vyvíjet, a ukazují, zda analytik správně uchopil jejich požadavky.
2. Pomáhají analytikovi zajistit, aby se zákazníci, pokud je jich více, shodli na tom, jaký systém se má zvolit.
3. Slouží designérům jako podklad pro návrh webových stránek.
4. Pomáhají vývojářům pochopit požadovanou funkčnost a technické požadavky na řešení.

V případě aplikace „Joaha“ se návrhy obrazovek používaly ve všech fázích analýzy problému. Wireframes sloužily jako zdroj požadavků ve fázi jejich sběru, pomáhaly při komunikaci se zákazníkem, usnadnily nalezení chyb ve specifikaci a nyní slouží jako podklad pro designera, který navrhne grafický vzhled stránky.

Návrh obrazovek jsem začala výběrem vhodného nástroje. Rozhodla jsem se použít Adobe Illustrator, což je vektorový grafický editor. Zvolila jsem jej namísto speciálních nástrojů pro tvorbu wireframes, jelikož mě neomezoval předpřipraveným vzhledem a poskytnul mi větší volnost při návrhu.

Ve výsledku vzniklo 19 obrazovek, které jsme se zákazníkem pro usnadnění komunikace očíslovali. Na obrázku 3.3 je zobrazen návrh stránky placeného videonávodu z pohledu uživatele bez členství. Její součástí jsou:

- Hlavní menu. Obsahuje odkazy na různé stránky aplikace, vyhledávání, tlačítko změny jazyka, odkaz na zprávy pro uživatele a na jeho profil.
- Vedlejší menu „Kategorie“. Umožní vyhledávání kategorií, témat a klíčových slov.
- Místo pro reklamu – jeden z požadavků zákazníka.
- Banner, který upozorní uživatele na to, že návod je ve zrychlené verzi.
- Samotný videonávod. Návod se přehraje zrychleně. Pod videem jsou odkazy na sdílení pomocí sociálních sítí.
- Diskuze a podobné návody.
- Footer, který je stejný na všech stránkách.

Wireframes nezachycují celý systém, pouze jeho nejdůležitější součásti, jako jsou například stránky návodů a online kurzů, přihlášení a registrace,

nákup nebo osobní profil uživatele. Stránky pro přidání příspěvků, správu komentářů nebo uživatelů poskytuje přímo CMS, a proto nejsou navrženy pomocí wireframes. Ostatní návrhy obrazovek jsou vzhledem k jejich velkému množství uloženy na přiloženém CD.

■ 3.6 Shrnutí kapitoly

V této kapitole jsem popsala proces specifikace a upřesnění požadavků, jejich analýzu a částečný návrh řešení. Po provedení těchto aktivit mám dostatečné množství podkladů k tomu, abych definovala požadavky na nástroj, který se použije pro vývoj, a zvolila právě takový.

The screenshot shows a website interface for 'JoAha'. At the top, there is a navigation menu with links: DOMŮ, FOTONÁVODY, VIDEONÁVODY, NÁVODY ZDARMA, ONLINE KURZY, and VÍKENDOVÉ KURZY. A search bar and a user profile icon are also present. Below the navigation, there are filters for 'KATEGORIE: Tuška, Pastelka ...', 'TÉMA: Vánoce, Velikonoce...', and 'KLÍČOVÁ SLOVA: člověk, růže...'. A yellow banner at the top of the content area reads: 'Tohle je zrychlená verze návodu. [Nákup členství](#)'. Below this is a large video player with a play button icon. Under the video player, there are social media sharing icons for Facebook, Google+, Twitter, and Instagram. The text 'Název návodu' is followed by a paragraph of placeholder text. Below that is a 'Diskuze' section with three comment icons and their corresponding text placeholders. A 'Rozbalit další' link is visible. Below the discussion is a section 'Může vás taky zajímat' with four placeholder images. On the right side of the page, there is a large vertical advertisement with the word 'reklama' written vertically. At the bottom of the page, there is a footer with links: 'Ochrana osobních údajů', 'O nás', 'Přehled členství', and 'Copyright'.

Obrázek 3.3: Placený videonávod pro uživatele bez členství

Kapitola 4

CMS systémy

Po analýze požadavků jsem zjistila, že většina funkcí aplikace „Joaha“ jako jsou komentování, přidání příspěvků, přihlášení, je často použitelná u aplikací, např. blogy a e-shopy. Proto jsem se rozhodla, že zkusím zjistit, jaké nástroje se používaly při tvorbě podobných systémů. Pro tento účel jsem použila plugin Wappalyzer pro prohlížeč Chrome¹. Tento plugin při návštěvě webové stránky detekuje platformu, na níž tato stránka běží, případně použité frameworky. Po návštěvě několika známých webových stránek jsem zjistila, že velké množství z nich běží na CMS systémech. Například stránky edX² mající funkcionalitu podobnou aplikaci „Joaha“ (poskytování placených a neplacených online kurzů) běží na platformě Drupal. Jako příklady blogů a e-shopů mohou sloužit web ČVUT³ – používá Drupal, populární servis Slevomat⁴ je postaven na Nette Framework, e-shop obchodu Cropp⁵ používá Magento.

Na základě těchto zjištění jsem se rozhodla použít CMS, který zjednoduší a urychlí vývoj tím, že použiju předpřipravené šablony a pluginy místo psaní funkcionality od začátku.

V následujících podkapitolách jsou popsány CMS systémy, vhodné pro vývoj aplikace „Joaha“, provedeno jejich srovnání a výběr jednoho z nich.

¹Dostupný z: <https://www.wappalyzer.com/>

²Dostupné z: <https://www.edx.org/>, navštíveno 2. ledna 2019

³Dostupný z: <https://www.cvut.cz/>, navštíveno 26. prosince 2018

⁴Dostupný z: <https://www.slevomat.cz/>, navštíveno 26. prosince 2018

⁵Dostupný z: <https://www.cropp.com>, navštíveno 26. prosince 2018

4.1 Definice CMS systémů

Podle Alexe Kempkense, jednoho z vývojářů Joomla!, CMS je aplikace, která poskytuje většímu počtu uživatelů s různými úrovněmi oprávnění možnost správy obsahu (celého nebo části), dat nebo informací o webových stránkách. Správa obsahu se týká vytváření, úpravy, archivace, publikování, distribuce obsahu, dat a informací [46]. Díky existenci open-source CMS systémů je vývoj webových aplikací levnější a rychlejší. Mnohé systémy jsou poskytovány s GNU GPL⁶ – jde o licenci, která dovoluje programátorům měnit kód podle svých potřeb. Díky tomu jsou aplikace postavené na těchto systémech zcela uzpůsobitelné dle potřeb zákazníka.

4.2 Výběr CMS systému pro vývoj aplikace Joaha

Ve fázi sběru požadavků jsem zjistila, že zákazník nemá požadavky na použité technologie. Z tohoto důvodu se při jejich výběru řídím svými požadavky jako vývojáře a schopností daných technologií vyhovět požadavkům na systém. Protože nemám zkušenost s vývojem pomocí CMS, volila jsem pro aplikaci „Joaha“ takový systém, který by splňoval mé základní požadavky a uspokojil potřeby zákazníka:

1. Musí být open-source. Tento typ software mi dovolí přidávat a upravovat kód podle mých potřeb. Pro zákazníka je provoz takové aplikace levnější a zároveň díky dalším požadavkům to nebude mít vliv na kvalitu systému.
2. Musí mít dostatečně velkou komunitu, abych se v případě problémů měla na koho obrátit.
3. Musí mít kvalitní dokumentaci. Tento požadavek má stejný účel jako předchozí. Výhodou pro zákazníka je to, že software s kvalitní dokumentací je snadno udržitelný.
4. Musí podporovat motivy, které bych mohla upravovat podle vlastních potřeb. S motivy bude vývoj aplikace rychlejší, jejich úprava umožní udělat libovolný design dle přání zákazníka.

Aby CMS systém splňoval druhý požadavek, musí být populární. Podle

⁶Dostupné z <http://www.gnugpl.cz/>

W3Techs byly v roce 2018 nejpoužívanějšími CMS systémy WordPress, Drupal a Joomla!^[28]. Tyto systémy splňují mé požadavky a nabízí ještě mnohem víc.

Všechny tři systémy jsou postaveny na PHP. Pro porovnání jsem se rozhodla zvolit si ještě jeden systém, který by byl napsán v jiném programovacím jazyce. Během studia jsem se seznámila s technologií Node.js, která je postavená na JavaScriptu. Node.js je vyvinut pro vysoce škálovatelné webové aplikace. Díky tomu, že nemá vlákna, ale používá asynchronní zpracování událostí, je v některých případech rychlejší a zvyšuje propustnost webového serveru^[49].

Jednou z nejpulárnějších alternativ WordPress, Drupal a Joomla! je Ghost ^[2]. Má dostatečně velkou komunitu a zároveň rozsáhlou a podrobnou dokumentaci. Další výhodou Ghost je jeho časté aktualizování⁷.

Nakonec jsem pro porovnání zvolila 4 CMS systémy: WordPress, Drupal, Joomla! a Ghost. Jejich výhody a nevýhody rozeberu podrobněji v následujících podkapitolách.

4.3 WordPress

Všechny informace o tomto CMS systému jsem čerpala z oficiálních stránek WordPressu⁸.

WordPress je jedním z největších systémů pro správu obsahu. Je to open-source framework pro vytváření webových aplikací, jako jsou blogy, e-shopy, statické stránky apod. Zásah do zdrojového kódu je nutný jen pro nejnáročnější uživatele – systém poskytuje velké množství pluginů a motivů. Podle oficiálních stránek WordPressu je 32% všech webů postaveno právě na tomto redakčním systému ^[53]. Dále se podíváme na výhody a nevýhody tohoto CMS.

Výhody WordPressu:

1. **Je to open-source software.** To umožňuje tvůrci webu zasáhnout do zdrojového kódu a upravit ho podle svých potřeb.
2. **Velká komunita po celém světě.** Podpora pro programátora, velké množství řešených problémů, návodů, best practices a diskuzí.

⁷Seznam posledních aktualizací: <https://github.com/TryGhost/Ghost/releases>

⁸Dostupné z: <https://cs.wordpress.org/>

3. **Více než 50 000 pluginů**[53]. Větší množství pluginů umožňuje přidání nových funkcí webové stránky bez napsání vlastního kódu. Pluginy umožňují přidání a správu e-shopu, online platbu, fóra, analytiku, propojení se sociálními sítěmi a mnohem více.
4. **Velké množství motivů.** WordPress poskytuje mnoho motivů, díky kterým lze rychle vytvořit atraktivní webovou stránku.
5. **Úprava motivů vlastním CSS.** Každou stránku lze doplnit vlastním stylem, což dává možnost vytvářet unikátní aplikace.
6. **Responzivita.** Motivy WordPressu podporují responzivní design, takže tvůrce webu nemusí vytvářet různé styly pro různá zařízení.
7. **Mnohojazyčnost.** Podle oficiálních stránek WordPress je tento systém přeložen do více než 160 jazyků [52]
8. **Úprava obsahu přes mobilní aplikaci** [54]. WordPress poskytuje mobilní aplikaci, přes kterou může uživatel přidávat a měnit obsah stránek.

Nevýhody WordPressu:

1. **WordPress se často aktualizuje.** Použité pluginy nemusí být kompatibilní s novou verzí, což může vyvolat výpadek webové aplikace a zvýšení nákladů na údržbu [31].
2. **Pomalé načtení webové aplikace.** Pro přidání doplňkové funkcionality je nutné instalovat pluginy, které zvyšují celkovou velikost aplikace. Kvůli tomu se stránky velmi pomalu nahrávají [33].

4.4 Drupal

Drupal je komplexní a flexibilní CMS platforma. Podle Ashleye Marshe jej lze popsat takto: „Drupal is the hardest but the most powerful CMS among all“, což v překladu znamená, že „Drupal je nejsložitější, ale nejrobustnější CMS mezi všemi“ [48]. Tento systém je vhodný pro tvorbu různých webových stránek, jako jsou e-shopy, online noviny a časopisy, personální blogy apod. Systém je škálovatelný a výkonný.

Výhody Drupal:

1. **Je to open-source software.** Programátor může měnit a doplňovat funkce podle svých potřeb.
2. **Nezávislý na OS⁹.** Drupal je možné instalovat na Linux, Windows nebo Mac OS X[30].
3. **Flexibilní struktura menu.** Drupal povoluje vytvoření mnohaúrovňového menu [6].
4. **Více než 16 000 pluginů [40].** Ty umožňují používat důležitou funkcionalitu, aniž by ji musel programátor psát sám.
5. **Velká škálovatelnost [40].**

Nevýhody Drupal [31]:

1. **Složitost vývoje pro nezkušeného uživatele.** Drupal vyžaduje pro vývoj technické znalosti, zatímco například WordPress a Joomla! umožní vytvořit webovou stránku i bez zkušeností v programování.
2. **Složitá instalace.** Instalace Drupal vyžaduje technické znalosti.

4.5 Joomla!

Informaci o tomto CMS systému jsem čerpala z oficiálních stránek Joomla! [50]

Joomla! se řadí mezi populární nástroje pro správu obsahu. Na této platformě dnes běží 2 miliony aktivních webových stránek [50]. Jedná se o vhodný nástroj pro tvorbu firemních stránek, online novin a časopisů, online rezervací, e-shopů, školních portálů atd. Systém poskytuje mnoho motivů a pluginů pro customizaci webových stránek.

Výhody Joomla!:

1. **Open-source software.** Každý vývojář může zasáhnout do kódu a upravit ho na základě svých požadavků.

⁹Operační systém

2. **Motivy.** Joomla! umožňuje používat motivy a měnit je podle vlastních potřeb.
3. **Multijazyčnost.** Joomla! podporuje 70 jazyků.
4. **Více než 8 000 pluginů .** Nadstavby umožňují přidávat nové funkce do webových stránek. Pomocí pluginů jde jednoduše přidat e-shop nebo posílání newsletterů.
5. **Verzování kontentu.** Joomla! na rozdíl od WordPressu ukládá verzi příspěvků, takže uživatel může kdykoliv vrátit provedené změny.
6. **Velká komunita po celém světě.**
7. **Posílání zpráv mezi administrátory.**
8. **LESS CSS.** Joomla! nabízí LESS preprocessor, který rozšiřuje funkcionalitu CSS například o proměnné.
9. **Responzivita.** Tento systém používá známý framework Bootstrap pro vytvoření responzivních stránek.
10. **Podpora reklamy.** Přidání reklamních bannerů je zabudovanou funkcí Joomla! a nevyžaduje instalaci pluginů.

Nevýhody Joomla! [14]:

1. **Potřeba velké výkonnosti serveru.** Stejná nevýhoda jako i u WordPress – pro velké aplikace potřebujeme výkonné technické vybavení, jinak se systém bude pomalu načítat.
2. **Problémy s kompatibilitou pluginů.** Některé pluginy nejsou kompatibilní, výsledkem je, že nemůžeme použít veškerou jejich funkcionalitu.

4.6 Ghost

Ghost je méně populární než WordPress nebo Joomla! [28], ale důvodem jeho zařazení do seznamu pro porovnání bylo použití jiného než PHP programovacího jazyka. Ghost na rozdíl od Wordpressu a Joomla! běží na Node.js serveru. Systém se hodí pro vytvoření blogů, online novin a časopisů. Má minimalistický design, který dovoluje soustředit se na vytvoření obsahu. CMS nenabízí pluginy, takže veškerou doplňkovou funkcionalitu je nutné vytvořit ručně.

Výhody Ghost:

1. **Open-source.** Programátorovi nic nebrání v tom, aby doplnil platformu o funkcionalitu, kterou potřebuje.
2. **Minimalistický design editoru [44].** Dovoluje soustředit se na vytvoření obsahu. Umožňuje napsání příspěvků v jazyce Markdown.
3. **Publikace obsahu podle harmonogramu [44].** Ghost dává možnost vytvořit harmonogram publikací příspěvků.
4. **Platforma běží na Node.js.** Je to zároveň výhoda i nevýhoda Ghost. Node.js je podporován hostingy mnohem méně než PHP. Naproti tomu Node.js nabízí možnost asynchronního zpracování požadavků, což zvyšuje propustnost webového serveru [49].
5. **Mobilní aplikace pro správu obsahu.** Mobilní aplikace je vyvinuta zatím jen na platformě Android [45]. Umožňuje přidávat a měnit příspěvky.

Nevýhody Ghost [10]:

1. **Absence pluginů.** Všechnu doplňkovou funkcionalitu pro webovou aplikaci je třeba dělat ručně.
2. **Nová platforma.** Ghost je nový systém, prozatím má v porovnání s WordPress a Joomla! málo uživatelů.

4.7 Porovnání CMS systémů

V této podkapitole porovnávám CMS systémy WordPress, Drupal, Joomla! a Ghost na základě informací uvedených v předchozích podkapitolách a údajů z oficiálních stránek systémů. Základními body srovnání jsou:

1. Požadavky na technologie
2. Cena a licence
3. Existence pluginů a motivů

4. CMS systémy

- 4. Podpora komerce
- 5. Možnost úpravy kódu

Ostatní vlastnosti představují výhody, ale nemají při rozhodování tak velkou váhu . Pro zpřehlednění jsem vytvořila srovnávací tabulku 4.1. Informace pro porovnání jsem čerpala z oficiálních stránek výrobců CMS systémů¹⁰.

¹⁰<https://wordpress.org/>, <https://www.drupal.org/>, <https://www.joomla.org/> a <https://ghost.org/>

	Wordpress	Drupal	Joomla	Ghost
Technologie				
Backend	PHP 7.2+ (5.2.4+). ¹	PHP 5.5.9+	PHP 5.6 nebo 7.0+	Node.js 7, 8 nebo 10. ⁵
Databáze	MySQL 5.6+(5.0+) ¹ nebo MariaDB 10.0+	MySQL 5.5.3+, MariaDB 5.5.20+, Percona 5.5.8+, PostgreSQL 9.1.2+, SQLite 3.4.2 +	MySQL 5.5.3+ nebo SQL Server 10.50.1600.1+ nebo PostgreSQL 9.1+	MySQL 5.5, 5.6, nebo 5.7
Server	Apache nebo Nginx ²	Apache, Nginx, Hiawatha nebo Microsoft IIS	Apache, Nginx nebo Microsoft IIS	Nginx
Licence a cena				
Open-source	ANO	ANO	ANO	ANO
Licence	GNU Public License	GNU Public License	GNU Public License	MIT License
Cena	Zdarma ³	Zdarma ³	Zdarma ³	Zdarma, placená verze ⁶
Vlastnosti				
Pluginy	ANO	ANO	ANO	NE
Komerce	ANO ⁴	ANO ⁴	ANO ⁴	NE
Motivy	ANO	ANO	ANO	ANO
Vlastní CSS	ANO	ANO	ANO	ANO
Úprava kódu	ANO	ANO	ANO	ANO
Responzivita	ANO	ANO	ANO	ANO
Multijazyčnost	ANO, 160 jazyků	ANO, 115 jazyků	ANO, 75 jazyků	NE (jen ručně)
SEO	ANO	ANO	ANO	ANO
Správa obsahu přes mobilní aplikaci	ANO	NE	NE	ANO
Verzování obsahu	NE	NE	ANO	NE

¹ Verze 5.2.4 není doporučena kvůli bezpečnosti

² Doporučené, jinak poběží na libovolném

³ Existují placené pluginy

⁴ Pomocí pluginů

⁵ Doporučený OS – Ubuntu 16.04 nebo 18.04

⁶ Placená verze poskytuje PaaS řešení

Tabulka 4.1: Porovnání CMS nástrojů

4.8 Výsledek porovnání

Podle porovnávací tabulky 4.1 vidíme, že Ghost nabízí mnohem méně vlastností než ostatní CMS. Nemá pluginy ani podporu komerce. Pokud bych zvolila Ghost, vývoj by trval delší dobu, což je pro zákazníka vždy nevýhoda.

Co se týká ostatních CMS systémů, jsou téměř na stejné úrovni. Poskytují pluginy, motivy, multijazyčnost, nejsou náročné na technologie backendu.

Nakonec jsem zvolila WordPress, protože zákazník má již jednu webovou aplikaci, která běží v tomto CMS. To znamená, že zákazník má pravděpodobně již kontakty na odborníky z WordPress, kteří by mohli pokračovat ve vývoji. Navíc zákazník bude muset absolvovat jen jedno školení, aby se naučil pracovat s WordPressem, a to mu dovolí ovládat obě aplikace. Nakonec v půlce projektu přibyl změnový požadavek, podle kterého bude aplikace propojená se stávajícími stránky zákazníka. Volba WordPressu jako platformy zjednoduší integraci systémů a jejich provoz (kapitola 8).



Kapitola 5

WordPress

V kapitole 4 jsem naznačila, že WordPress je jedním z nejpobulárnějších CMS systémů. Slouží pro zjednodušení vývoje a podpory webových aplikací různých druhů. Protože podle porovnání WordPress nejvíce odpovídá požadkům a potřebám zákazníka, věnuji v této kapitole pozornost hlavně jeho základním kamenům.

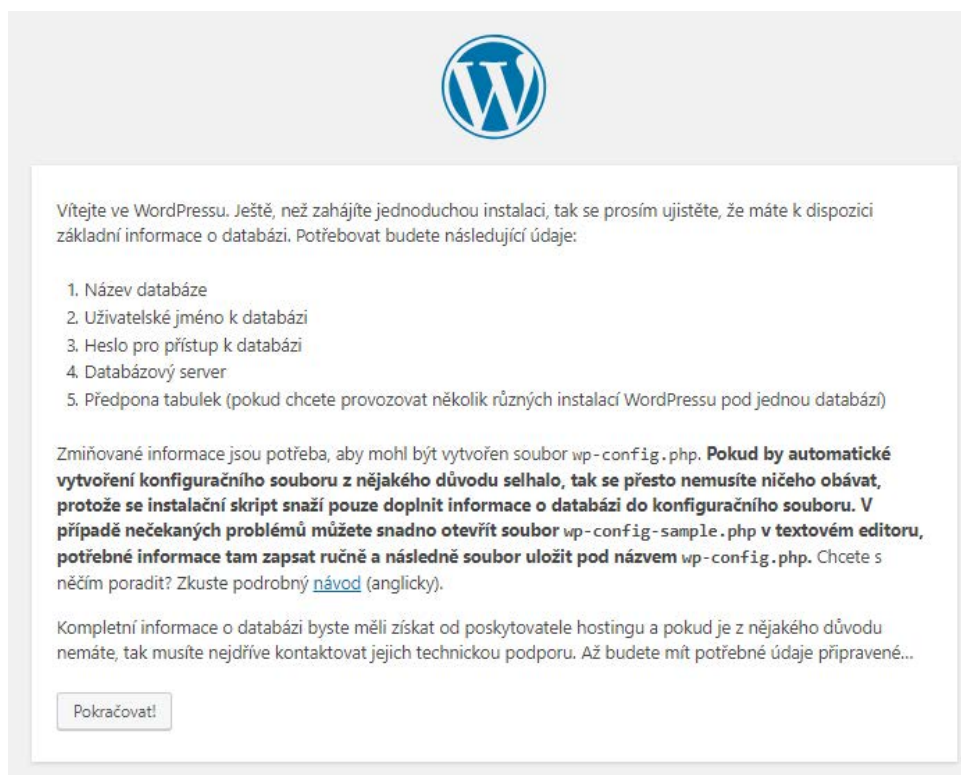
Popis základní funkcionality CMS považuji za důležitý pro neznalé uživatele. Setkala jsem se s WordPressem poprvé, proto popíšu jeho technologie z pohledu začátečníka v programování pomocí tohoto systému. Většinu informací jsem čerpala z tutoriálů, dokumentace WordPressu nebo prostým zkoušením.

V dalších kapitolách budu používat terminologii WordPressu, která často nemá překlad do češtiny. Nepřeložitelné pojmy jsou vysvětleny ve slovníku na konci práce.



5.1 Technologie a instalace

WordPress je napsán na platformě PHP s použitím webových technologií typu JavaScript (knihovna JQuery) a CSS. Instalace systému je snadná a je přizpůsobena i pro neznalého uživatele. Postup instalace je následující:



Obrázek 5.1: Popis instalace od WordPress

1. Uživatel si stáhne balíček WordPress z oficiálních stránek systému [23]. Jak jsem psala dříve, WordPress je dostupný ve 160 jazykových mutacích. Jazyk se pak dá upravit v nastaveních systému.
2. Balíček se vloží do adresáře na hostingu a otevře se hlavní stránka aplikace.
3. Uživatel uvidí přivítání od WordPressu s detailním popisem instalace - obrázek 5.1. K tomu, aby založil web, musí ručně vytvořit databázi (nebo o ni poprosit provozovatele hostingu).
4. Zadáním základních údajů (informací o databázi, účtu administrátora nové webové aplikace) uživatel dokončí instalaci.

Podle stránek WordPressu trvá instalace 5 minut. Celý popis lze najít v Codexu, kde se nachází online dokumentace a uživatelský manuál [12].

Samotný balíček WordPress aplikace se skládá ze 4 adresářů:

- **Root adresář.** Tady se nachází hlavně konfigurace aplikace - přístup k databázi, administrátorský účet, nastavení apod.

- **wp-admin.** Veškerá data, která se týkají administrátorského prostředí aplikace. To jsou například soubory pro správu médií, uživatelské nastavení, stránka editoru.
- **wp-content.** Tento adresář je nejdůležitější pro uživatele, protože se zde nachází veškerá data pluginů a vzhledů.
- **wp-includes.** Ostatní soubory z WP Core, tzn. CSS soubory, API apod.

5.2 Plocha administrátora

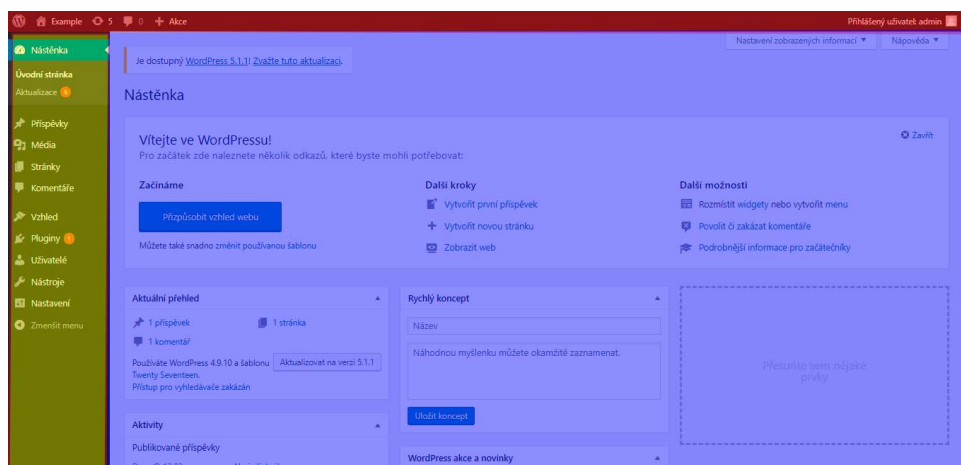
Nejvyšší role ve WordPressu je administrátor. Uživatel s těmito oprávněními má přístup ke všem nastavením aplikace včetně správy uživatelů, příspěvků, stránek, pluginů a vzhledů. Proto musí jít o osobu, která je seznámena s technologií, ideální volbou je tedy vývojář nebo proškolený uživatel.

Plocha administrátora je část webové aplikace, která slouží k přizpůsobení vzhledu a funkcionality a k naplnění stránek obsahem. Pro přihlášení k této části webu je nutné použít adresu `/wp-admin/`.

Stránka se skládá ze 3 částí (obrázek 5.2):

- **Admin Bar.** Nachází se v horní části obrazovky a poskytuje rychlou navigaci do webu, uživatelského profilu, stránek WordPressu a úpravy vzhledu.
- **Admin Panel.** Levé menu, které slouží pro veškerou navigaci v administrátorském účtu. Poskytuje přístup k pluginům a šablonám, nastavením, editoru a nástěnce.
- **Hlavní část.** Mění se podle toho, kde se zrovna nachází uživatel.

Různé pluginy přidávají do Admin panelu své položky, aby urychlily přístup k dalším nastavením.



Obrázek 5.2: Plocha administrátora. Červená – admin bar, žlutá – admin panel, modrá – hlavní část.

5.3 Editor

Od verze 5.0 nabízí WordPress uživateli dva druhy editoru: HTML (klasický) a vizuální (Gutenberg) [42]. Přepínač mezi editory se nachází v nastaveních a na samotné stránce editace.

Klasický editor je určený přímo pro práci s HTML značkami. Nabízí uživateli i tlačítka pro přidání odstavců, odkazů apod., což je pro začátečníka velká výhoda. Nevýhodou je, že uživatel nevidí výsledek své práce okamžitě, zobrazí se až po publikaci (případně náhledu) příspěvku nebo stránky. Tvorba responzivních webových stránek vyžaduje buď minimální znalost HTML a CSS nebo použití pluginů typu Elementor nebo Divi (pluginy pro vizuální tvorbu rozvržení webu).

V současné době nabízí WordPress editor Gutenberg, který je WYSIWYG¹ typu. Pro tvorbu stránek používá bloky, což jsou jak jednoduché části typu obrázků a odstavců, tak i rozvržení do sloupců a složitější komponenty. Gutenberg je drag-and-drop editor s moderním designem, takže se v něm vyzná i začátečník. Nevýhodou je, že se uživatelé velmi často setkávají s chybami, což je zřejmé z jejich komentářů na stránkách projektu [34].

¹What You See Is What You Get - co vidíš, to dostaneš.

5.4 Pluginy a vzhledy

Při prvním spuštění nabízí WordPress uživateli předinstalované vzhledy „Twenty Seventeen“, „Twenty Fifteen“ a „Twenty Sixteen“. Uživatel si může nainstalovat jiné šablony pomocí menu z Admin panelu Vzhled→Šablony. Šablony mění větší část frontendu a jejich primárním úkolem je správně použít WP Core pro zobrazení informací uživateli. Při volbě vzhledu je třeba dávat pozor na pluginy, které budou u webových stránek použity. Jedná se nejen o kompatibilitu, ale i o možnost přizpůsobení té či jiné částí webu.

Pluginy jsou balíčky doplňující nebo měnící funkcionalitu WordPress. Nepracují samostatně, ale většinou používají funkce z WP Core. Slouží primárně k rozšíření funkcionality webu bez přidání vlastního kódu. Při instalaci pluginů je třeba dávat pozor na jejich vzájemnou kompatibilitu, podporovanou verzi WordPressu a požadovanou verzi PHP a MySQL.

Nezbytným požadavkem na technologie byla možnost měnit kód a vzhled aplikace podle vlastních potřeb. Přestože vývojář vidí veškerý kód WordPressu, není doporučeno do něho nějakým způsobem zasahovat. Jedním z důvodů je, že při aktualizaci systému, šablon a pluginů se některé složky přemažou a uživatel může ztratit veškeré své změny. Proto nabízí WordPress uživatelům možnost vytvoření Child Theme. Jedná se o šablonu, odvozenou z jiné šablony WordPress. Pokud uživatel vytvoří prázdnou child theme, není patrná žádná změna. Následník používá veškerý kód (vzhled a chování) rodičovské šablony, dokud uživatel nepřepíše potřebné soubory. Například pokud vývojář chce změnit výchozí rozvržení příspěvků, zkopíruje soubor **single.php** do child theme a následně ho upraví podle svých potřeb [13].

5.5 Aktualizace a kompatibilita

Důležitou částí provozu aplikace je aktualizace jejích modulů. V případě WordPressu se aktualizují samotný CMS systém, šablony a pluginy. Aktualizace je pro web běžící na tomto systému nezbytná. Jedná se nejen o podporu, ale také o bezpečnost, rychlost, opravu chyb a kompatibilitu. Kroky, které je třeba udělat téměř před každou aktualizací, zvláště pokud se jedná o samotný CMS:

- Zvážit aktualizaci.

- Super Admin - super administrátor - má nejvyšší oprávnění, a to v celé síti webů (pokud se používá WordPress Multisite)
- Admin - administrátor - má nejvyšší oprávnění v rámci dané webové aplikace.
- Editor - redaktor - má právo spravovat příspěvky a komentáře.
- Author - autor - má právo publikovat a spravovat své příspěvky.
- Contributor - spolupracovník - může psát příspěvky, ale nesmí je publikovat (obvykle je posílá ke schválení redaktorovi nebo administrátorovi).
- Subscriber - návštěvník - běžný uživatel, který nemá skoro žádná oprávnění.

Úpravou kódu nebo pomocí pluginů lze přidat nové role, měnit jejich oprávnění a přiřazovat role uživatelům.

Když se uživatel dostane na webovou stránku poprvé, není přihlášený a nemá tedy žádnou roli. Ta je mu přiřazena až po registraci. Příslušnou roli uživatele definuje administrátor, který ji následně může také měnit.

5.6.2 Příspěvky a vlastní typy

Zakladatelé WordPressu prezentují tento CMS hlavně jako publikační systém [35], proto základními prvky jsou příspěvky². Podle Codexu se tím rozumí typ příspěvku, a to například stránka, příloha, navigační menu nebo dokonce vlastní CSS. Všechny příspěvky mají společné prvky jako název, slug³, povolená taxonomie apod. Pro potřeby uživatelů poskytuje WordPress funkci `register_post_type`, která umožní uživateli vytvářet vlastní druhy příspěvků. Například v případě aplikace „Joaha“ jsem vytvořila fotonávod, videonávod, obrázek. Pluginy také vytváří vlastní druhy příspěvků jako produkt (Woocommerce) nebo kurz (LifterLMS).

Výhodou existence různých typů příspěvků je to, že administrátor může spravovat přístup uživatelů ke čtení, editaci, publikaci u jednotlivých typů

²anglicky - Post

³Podle slovníku WordPress, je slug slovo nebo několik slov popisujících příspěvek, které se dají použít v URL. Například, v odkazu www.wordpress.org/post/welcome-post je slugem „welcome-post“.

Kapitola 6

Průběh implementace aplikace

V této kapitole bych ráda popsala samotný průběh implementace aplikace „Joaha“. Vzhledem k tomu, že jsem používala CMS systém, budu se věnovat hlavně popisu práce ve WordPressu, a to výběru vývojového prostředí, přizpůsobení kódu, použitým pluginům apod. Účelem této kapitoly je takový popis implementace, který by ulehčil neznámému čtenáři seznámení se s WordPressem a který by mohl být jistou podobou tutoriálu.

Samotnou implementaci jsem rozdělila do 5 částí:

1. Seznámení se s prostředím, instalace IDE, čtení dokumentace.
2. Implementace požadavků s prioritou MH.
3. Implementace požadavků s prioritou SH.
4. Implementace CH požadavků.
5. Úprava vzhledu, design.

Implementace požadavků s různými prioritami probíhaly obdobně, proto jsem sjednotila jejich popis do jedné podkapitoly.

6.1 Vývojové prostředí

Protože WordPress používá PHP a MySQL, potřebovala jsem pro vývoj lokální server. Na začátku jsem zvolila AMPPS[29] server, který nabízel zpracování PHP scriptů, MySQL databázi a zcela automatizovanou instalaci WordPressu. Po několika týdnech použití AMPPS jsem se setkala s problémem zastaralé verze PHP. Během vývoje se snažím používat nejnovější verze WordPressu, pluginů a šablon, což server AMPPS neumožňoval. Kvůli tomu jsem instalovala server XAMPP[39], který nabízí nejnovější verze PHP.

Pro úpravu PHP souborů se dá použít libovolný textový editor, ale z vlastní zkušenosti vím, že vývoj v IDE je mnohem pohodlnější a rychlejší. Zvolila jsem komerční řešení od JetBrains - PhpStorm[19], který nabízí mnoho výhod. Za prvé poskytuje navigaci kódem, která pomáhá zorientovat se v různých pluginích a v samotném WP Core. Za druhé vyhodnocuje některé části kódu a rovnou upozorňuje na chyby a překlepy. Největší výhodou pro mě bylo full-text vyhledávání v kódu, které mi umožnilo najít části, jež mě současně zajímaly.

Co se týká samotného seznámení s WordPressem, v předchozích kapitolách jsem naznačila, že CMS poskytuje online dokumentaci Codex, která kromě popisu funkcí obsahuje i některé příklady použití. Nezbytnou částí seznámení se s WordPressem bylo čtení fór a blogů, například, WPBeginner [36].

6.2 Průběh implementace požadavků

Implementaci požadavků lze rozdělit do dílčích bloků:

1. Přidání funkcionality pomocí pluginů. Tuto část detailněji popíšu v kapitole 6.3.
2. Přidání funkcionality pomocí vlastního kódu (podkapitola 6.4).
3. Úprava funkcionality WordPressu a pluginů.
4. Úprava šablony – vzhled a funkcionality.

Pro přidání vlastní funkcionality jsem vytvořila child theme - šablonu, která

upravovala šablonu Storefront. Důvodem bylo to, že během implementace jsem zjistila, že pouhá úprava CSS mi nestačí.

Hlavními součástmi šablony jsou soubory `function.php` a `style.css`. Jak napovídají rozšíření souborů, `function.php` se používá pro úpravu php scriptů.

Protože WordPress je populární CMS systém a používá jej velké množství lidí, poskytuje velké API pro úpravy funkcionality a vzhledu. Hlavními funkčními prvky WordPressu jsou akce (actions), filtry (filters) a zkrácené kódy (shortcodes).

- Akce je funkce, která přidává data nebo upravuje funkce WordPress. Nemá návratovou hodnotu. Například další kus kódu přidává funkcionality pro sledování popularity příspěvku (při otevření příspěvku zavolá funkci pro zvětšení počtu návštěv). Akce se provede při vykreslení hlavičky stránky (to definujeme přidáním akce do `wp_head`).

```
function wpb_track_post_views ( $post_id ) {
    if ( ! is_single ( ) ) return;
    if ( empty ( $post_id ) ) {
        global $post;
        $post_id = $post->ID;
    }
    wpb_set_post_views( $post_id );
}
add_action( 'wp_head', 'wpb_track_post_views' );
```

- Filtry – funkce, které jsou hodně podobné akcím, s tím rozdílem, že data zpracovávají, mění a vrací zpět do rodičovské funkce. V dalším příkladě mění filtr obsah tím, že do něho přidá Javascript pro zrychlení videa. Filtr se liší od akce tím, že jako parametr přijme obsah (`$content`) a jako výsledek ho vrátí.

```
function restricted_videonavod ( $content ) {
    $content .= '<script>
        document.addEventListener ( "DOMContentLoaded",
        function ( ) {
            var v = document.querySelectorAll ( "video " );
            v [ 0 ].playbackRate = 3;
        } );
    </script>';
    return $content;
}
```


6.3.1 Online kurzy

Pro vytvoření online kurzů jsem se rozhodla použít LMS – systém pro podporu výuky. Podobné systémy nabízí hotové řešení pro online kurzy, kvízy či testy na webových stránkách. V první verzi bude aplikace „Joaha“ poskytovat jen základní funkčnost pro online kurzy. Při vývoji jsem ale počítala s tím, že se aplikace bude s časem rozšiřovat, proto jsem hledala pluginy se stabilní podporou a funkcionalitou navíc.

Pro LMS plugin jsem stanovila takové minimální požadavky, které vychází z analýzy a wireframů:

- Možnost vytvoření kurzu bez domovské stránky (podle wireframes při otevření kurzu se rovnou otevírá lekce tohoto kurzu).
- Možnost dělení lekcí do sekcí.
- Úprava vzhledu kurzu.
- Zobrazení komentářů ke kurzu na stránkách jednotlivých lekcí.
- Zobrazení informace o kurzu na stránkách jednotlivých lekcí.
- Podpora, aktualizace a dokumentace.

Po vyhledávání pluginů v uložišti WordPress jsem zvolila 4 pluginy, které jsem porovnávala mezi sebou, a to LifterLMS [17], CoursePress [5], LearnPress [16] a WP-Courses [37]. Každý z nich měl své výhody a nevýhody. Pluginy CoursePress a Wp-Courses jsem neotestovala do konce, protože jsem je odmítla na začátku testování. První z nich měl podle komentářů velmi špatnou podporu a málo časté aktualizace. Druhý jsem musela odmítnout kvůli absenci dělení lekcí do sekcí.

Celé porovnání je uvedeno v tabulce 6.1.

	LifterLMS	CoursePress	LearnPress	WP-Courses
Absence domovské stránky kurzu	Ne	Ne	Ne	Ano
Dělení lekcí do sekcí	Ano	Ne	Ano	Ne

Úprava vzhledu	Snadná. Vystavěná funkcionální pro úpravu šablon.	Nebylo zkoušeno. Plugin jsem odmítla kvůli absenci podpory.	Mírně složitá. Přístup jen k CSS souborům.	Nebylo zkoušeno. Kvůli absenci dělení lekcí do sekcí plugin neodpovídal požadavkům (spor s wireframes)
Komentáře pro kurz	Kurz a lekce jsou typy příspěvků, takže mají od WordPress podporu komentářů. Pro zobrazení komentářů kurzu u jeho lekcí byl nutný zásah do kódu	Nebylo zkoušeno. Plugin jsem odmítla kvůli absenci podpory	Kvůli složitosti přizpůsobení (nejsou šablony, přizpůsobení jen pomocí css) je složité přidat tuto funkcionální ručně a zachovat udržovatelnost pluginu.	Je možné přidat tuto funkcionální ručně.
Informace o kurzu	Informace o kurzu se získala zásahem do kódu.	Nebylo zkoušeno. Plugin jsem odmítla kvůli absenci podpory	Kvůli složitosti přizpůsobení (nejsou šablony, přizpůsobení jen pomocí css) je složité přidat tuto funkcionální ručně a zachovat udržovatelnost pluginu.	Lze přidat zásahem do kódu.
Podpora, dokumentace	Dokumentace je velmi široká, LifterLMS má velkou podporu pro vývojáře (velké množství shortcutů, akcí a filtrů pro přizpůsobení)	Podle komentářů uživatelů, počtu řešených chyb a frekvenci aktualizací jsem rozhodla, že podpora pluginu není důveryhodná.	Plugin poskytuje málo možností úpravy vzhledu a funkcionality.	Plugin jsem nehodnotila z toho pohledu, protože jsem ho odmítla dříve.

Vlastnosti navíc	Vestavěná funkcionality pro nákup online kurzu a omezení obsahu pro nečleny.	Nákup kurzů, velké množství platebních bran, certifikáty a kvízy	Prodej kurzů, platební brány lze získat z placených pluginů.	Funkcionality rozšiřují pluginy.
Vyhodnocení	LifterLMS má velkou a použitelnou dokumentaci, je připraven pro změny. Poskytuje šablony pro úpravu vzhledu skoro veškerého obsahu. Obrovskou výhodou je vestavěná funkcionality pro nákup kurzu a omezení obsahu pro nezapsané uživatele.	Většinu požadavků jsem nezkoušela kvůli tomu, že podle hodnocení a komentářů na moment srovnání plugin neměl podporu vývojářů ani nebyl aktualizován víc než 3 měsíce.	Plugin se mi zdál být nepoužitelný pro aplikaci „Joaha“ kvůli složitému přizpůsobení vzhledu.	Přestože absence domovské stránky kurzu byla jedním z požadavků, vedlo to ke složitějšímu přidání informací o kurzu a společných komentářů. Navíc plugin nemá vestavenou funkcionality pro dělení lekcí do sekcí.

Tabulka 6.1: Porovnání pluginů - online kurzy

Po porovnání jsem se rozhodla použít plugin LifterLMS, protože nejvíc odpovídá minimálním požadavkům a je velmi flexibilní vzhledem k přizpůsobení. Navíc má velmi dobrou dokumentaci a API pro vývojáře [15].

■ 6.3.2 Členství

V případě členství jsem nějaký čas vybírala mezi vlastní implementací a použitím pluginu. V případě WordPressu by se dalo členství implementovat jako role uživatelů nebo jako položku v meta datech. Nevýhodou vlastního

řešení je absence podpory a horší udržovatelnost. Proto jsem nakonec přistoupila k volbě pluginu, který by poskytoval základní funkčnost, ale nebyl příliš přeplněný funkcionalitou navíc. Podle analýzy jsem stanovila následující požadavky na plugin:

- Uživatel může mít více členství najednou.
- Členství se dá přidávat ručně z administrátorské plochy.
- Omezení obsahu pro nečleny - spíše přání (dá se udělat ručně).
- Neomezený počet členství.
- Informace o stavu členství se dá získat přes kód.
- Členství je časově omezené.

Pro porovnání jsem zvolila pluginy Groups [11], WooCommerce Membership [32], Paid Membership Pro [18] a WP-Members [38]. Jejich srovnání lze najít v tabulce 6.2.

	Groups	WooCommerce Membership	Paid Memberships Pro	WP-Members
Cena	Minimální požadovaná funkcionalita je zdarma. Pro prodej členství je třeba zakoupit plugin.	Celý plugin je placený.	Plugin je zdarma, ale má více než 60 placených rozšíření	Plugin je zdarma, integrace jsou v podobě placených pluginů
Více členství najednou	ANO	ANO	ANO	ANO
Přidání členství ručně	ANO	ANO	ANO	Pouze vlastní implementace, existuje jenom PHP akce.
Neomezený počet členství	ANO	ANO	ANO	ANO
Časově omezené členství	Pomocí pluginu.	Ano	Ano	Ano

Omezení obsahu	Ano, neplacená funkcionalita, pomocí shortcodů. Lze zavřít libovolnou část příspěvku.	Lze omezit jen přístup k typům příspěvků.	Ano, pomocí shortcodů.	Lze omezit jen přístup k typům příspěvků.
Přístup přes kód	Ano, je přístup pomocí shortcodů.	Ano, pomocí filtrů a akcí.	Ano, pomocí filtrů a akcí.	Ano, pomocí filtrů a akcí.
Prodej členství	Propojení s WooCommerce pomocí pluginu.	Ano, zabudovaná funkcionalita.	Ano, plugin nabízí 6 zabudovaných platebních bran. Ostatní platební brány lze přidat pomocí pluginů	Pomocí pluginu.

Tabulka 6.2: Porovnání pluginů - členství

Po porovnání jsem vybrala plugin Groups. Má veškerou požadovanou funkcionalitu a také přehlednou dokumentaci. Nenabízí zbytečnou funkcionalitu a jeho použití je jednoduché.

6.3.3 Vzhled a rozvržení, šablony

Podle návrhu v aplikaci „Joaha“ tvoří fotonávody, na rozdíl od ostatních typů příspěvků, velmi složité rozložení. Podle mých předpokladů nebude člověk, který bude přidávat návody, seznámen s HTML ani s CSS. Proto je nutné co nejvíce automatizovat vytvoření takového rozložení. Nabízí se využití různých WYSIWYG editorů, které uživateli rovnou ukazují, jak bude vypadat jeho příspěvek. Další možností bylo vytvořit vlastní šablonu příspěvku, do které by pak uživatel jen dával obrázky a text.

Hledala jsem tedy plugin, který by umožňoval buď vytvořit šablonu příspěvku nebo přidat složité rozvržení bez znalostí HTML. Minimální požadavky na takový plugin:

- Možnost vytvoření šablon příspěvků.
- Vytvoření složitého rozvržení (sloupce, víceúrovňové zanoření).
- Podpora shortcutů (tento požadavek zaručuje možnost použití jiných pluginů, například pro omezení přístupu k obsahu, přidání sdílení přes sociální síť apod.)
- Uživatel nemusí pracovat s HTML ani CSS – ideálně WYSIWYG editor.

Vybrala jsem 3 pluginy, které na první pohled splňovaly všechny požadavky: Elementor [7], Brizy [4] a Gutenberg (dříve plugin, teď vestavěný editor WordPressu) [25]. Jejich srovnání je vidět v tabulce 6.3.

	Elementor	Brizy	Gutenberg
Vytvoření šablon	Ano, pomocí pluginu	Nebylo zkoušeno, hned po instalaci se objevil problém – nekompatibilita s WordPress editorem (Gutenberg).	Nelze vytvořit šablonu celého příspěvků, ale pomocí pluginu Block Labs lze přidat vlastní blok (v případě aplikace „Joaha“ – krok fotonávodů)
Složitě rozvržení	Ano, rozvržení pomocí bloků, lze rozšířit nabídku bloků pomocí pluginů.	Ano	Ano
Podpora shortcutů	Obecně ano, ale nepodporuje zavírací shortcody (typu <code>[code]obsah[\code]</code>). Takový druh shortcutů používá plugin Groups, který jsem zvolila pro implementaci členství, proto jsem měla plugin Elementor odmítnout.	Nebylo zkoušeno, hned po instalaci se objevil problém – nekompatibilita s WordPress editorem (Gutenberg).	Podporuje všechny typy shortcutů
WYSIWYG editor	ANO	ANO	ANO
Důvod odmítnutí	Není kompatibilní s pluginem Groups – nefunguje zkrácený zápis	Není kompatibilní s editorem Gutenberg – je důvodem bugu.	–

Důvod zvolení	–	–	Výchozí editor, díky čemuž je vysoká kompatibilita. Doplněný pluginem Block Lab splňuje požadavek na šablony příspěvků.
---------------	---	---	---

Tabulka 6.3: Porovnání pluginů – tvorba vzhledů

Bohužel na tomto kroku jsem se poprvé setkala s nekompatibilitou pluginů. Elementor nepodporoval uzavírající se shortcody, a Brizy způsoboval špatné fungování editoru WordPress. Nakonec jsem zvolila Gutenberg, doplněný pluginem Block Labs, který přidává nové bloky (v případě aplikace „Joaha“ – krok fotonávodů).

6.4 Vlastní úpravy kódu

Některá požadovaná funkcionalita musela být implementována ručně kvůli absenci potřebných pluginů. Většinou se jednalo o jednoduché funkce nebo o přidání některé funkcionality do pluginů. Vlastním kódem jsem přidala například integraci pluginu BuddyPress a User Bookmarks, WooCommerce apod (jedná se o přidání záložek Oblíbené, Členství do profilu uživatele). Někdy bylo nutné opravit chyby v pluginech nebo pozměnit vzhledu šablony. V této podkapitole uvedu krátký popis všech úprav (detailní popis se nachází v dokumentaci aplikace):

1. Přidání funkcionality

- a. Tlačítko „Více“ pro nahrání komentářů místo výchozího stránkování.
- b. Funkce pro použití prvního obrázku fotonávodů místo obrázku náhledového.
- c. Automatizace skrývání obsahu pro nečleny (shortcode, který automaticky určuje členství uživatelů, jež budou mít přístup k příspěvku, a to na základě kategorií).

- d. Zrychlení videonávodu pro nečleny.
 - e. Vyhledávací menu „Kategorie“ (hledání příspěvků na základě kategorií, štítků a témat).
 - f. Změna viditelnosti složek v oblíbeném.
2. Oprava chyb
- a. Špatné zobrazení veřejných záložek v pluginu User Bookmarks (ve verzi 3.4 se zobrazovaly jen složky přidané administrátorem, nikoli uživatelem).
3. Úprava vzhledu
- a. Úprava vzhledu komentářů.
 - b. Přidání vyhledávacího pole a notifikací do menu.
4. Integrace
- a. Nové záložky v profilu uživatele (členství, oblíbené, vložení příspěvků apod.)
 - b. Přidání notifikací BuddyPress při různých akcích WordPressu (například vložení uživatelem příspěvků).

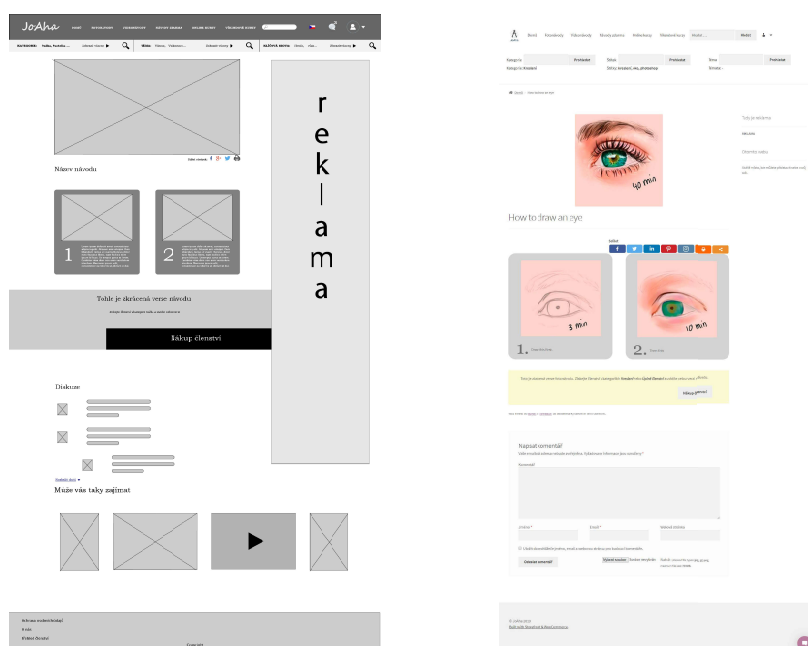
Jak jsem zmínila dříve, pro implementaci jsem používala akce, filtry a short-cody, které jsem přidávala do souboru `functions.php` patřícího vytvořené child theme.

6.5 Vzhled aplikace

Po domluvě se zákazníkem dělala design aplikace třetí strana. Během implementace jsem upravovala vzhled tak, aby byl podobný wireframům (hlavně rozložením). V době odevzdání práce nebyl finální design připraven, proto zůstává aplikace prozatím podobná wireframům. Na obrázku je vidět, jak vypadá aplikace do úpravy vzhledu podle designu. Pro porovnání uvedu i příslušný wireframe (obrázek 6.1).

U některých stránek (například profil uživatele – obrázek 6.2) se rozložení velmi liší od wireframů, jelikož jsem při implementaci usoudila, že takové rozložení bude pro uživatele pohodlnější a zároveň pochopitelnější.

Pro implementaci jsem zvolila šablonu Storefront, která je kompatibilní s WooCommerce a má minimalistický a jednoduchý design. V případě, že bych



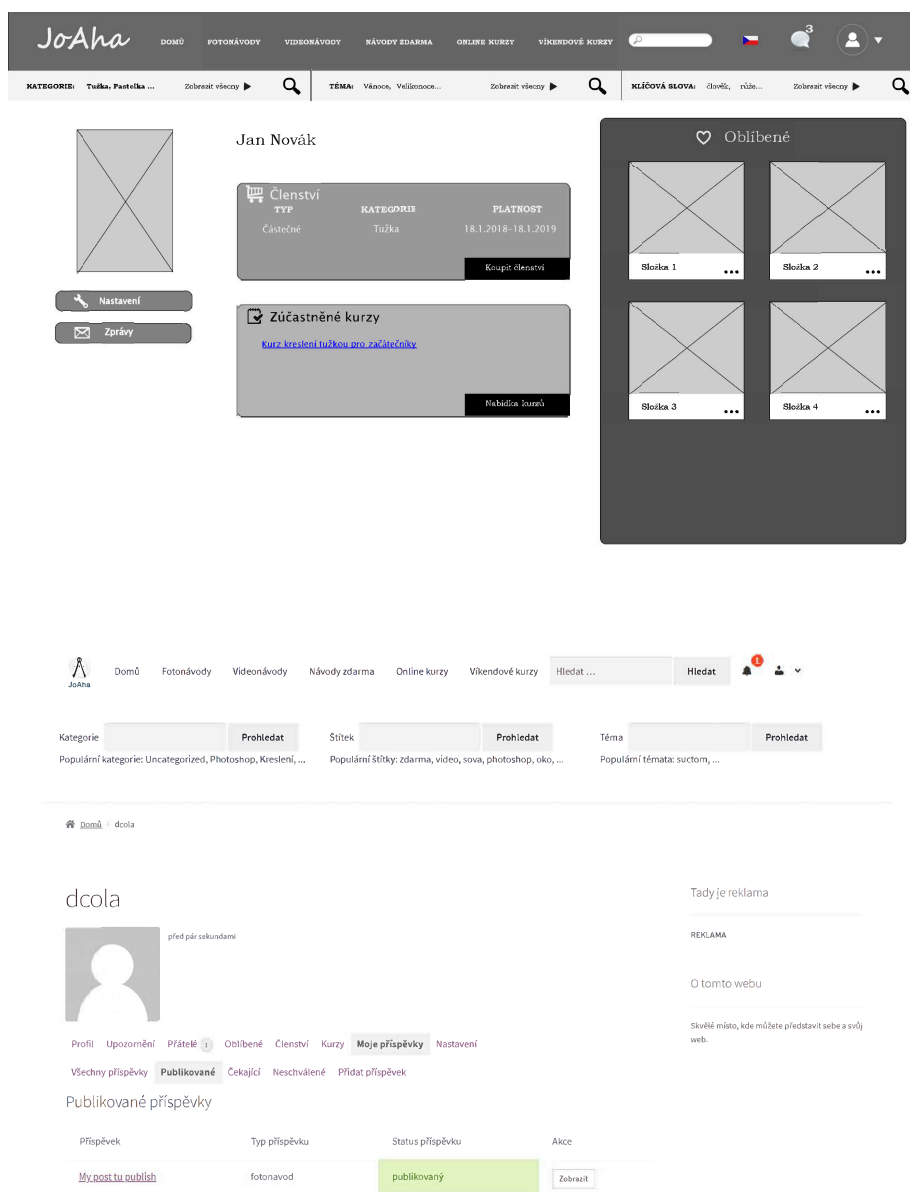
Obrázek 6.1: Neplacený fotonávod – wireframe a design

měla design aplikace od začátku vývoje, zvolila bych šablonu, která se mu nejvíce podobá. V mé situaci by si změna šablony v takto pokročilé etapě vývoje vyžádala úpravu velkého množství funkcí, což by znamenalo značnou ztrátu času.

6.6 Shrnutí procesu implementace

V této kapitole jsem popsala průběh implementace požadavků. Detailnější informaci o fungování systému lze najít v uživatelském manuálu na příloženém CD, kde se také nachází kompletní dokumentace aplikace.

6. Průběh implementace aplikace



Obrázek 6.2: Profil uživatele - wireframe a design

Kapitola 7

Testování

Pro ověření funkčnosti aplikace „Joaha“ jsem zvolila dvě úrovně testování: FAT ¹ a UAT ² testy.

FAT testy se prováděly po implementaci každého požadavku (případně skupiny požadavků). Někdy bylo kvůli problémům s kompatibilitou pluginů nutné provádět ověření funkčnosti opakovaně (například po určité konfiguraci pluginu UserBookmarks nastává problém s publikací příspěvků pomocí editoru Gutenberg. Výběr správné konfigurace vyžaduje opakované testování).

Během konzultací probíhaly průběžné UAT testy. Byly testované hlavní části aplikace, a to implementace požadavků MH a SH. Po testování je vidět, že zadání bylo pochopeno správně, jelikož změnové požadavky, popsané v kapitole 9.2, se týkají jen doplnění funkčnosti.

Finální UAT testování zatím nebylo provedeno z důvodů absence designu aplikace. Připravila jsem testovací scénáře, podle kterých by zákazník mohl ověřit funkčnost celé aplikace. Testování se rozdělí do dvou částí: testování administrátorské části a uživatelské. V této kapitole uvedu několik scénářů pro uživatelské testování. Pomocí těchto průchodů se dá zkontrolovat hlavní funkčnost aplikace.

¹funkční testy

²uživatelské akceptační testy

7.1 Testovací scénáře – administrátorská část

7.1.1 Scénář A1

Cíl: vložení různých typů příspěvků (online kurz, fotonávod, videonávod, obrázek). Kontrola toho, že v editoru nechybí možnost přidání důležitých informací typu taxonomie.

Průchody:

1. Uživatel se přihlásí jako administrátor.
2. Uživatel přidá fotonávod. Ověří, zda je možné přidat kategorie, štítky, témata.
3. Uživatel udělá fotonávod placeným a skryje jednu jeho část od uživatelů bez členství.
4. Uživatel se podívá na kontrolu před publikací a ověří, zda mu systém připomíná, aby:
 - a. definoval, zda je návod zdarma
 - b. skryl nějakou část návodu
 - c. přidal taxonomii.
 - d. přidal náhledový obrázek
5. Uživatel přidá videonávod. Ověří, zda je možné přidat kategorie, štítky, témata.
6. Uživatel udělá videonávod placeným.
7. Uživatel se podívá na kontrolu před publikací a ověří, zda mu systém připomíná, aby:
 - a. definoval, zda je návod zdarma
 - b. přidal taxonomii.
 - c. přidal náhledové video.
8. Uživatel vloží obrázek. Otestuje, zda je možné přidat kategorie, štítky, témata.
9. Uživatel se podívá na kontrolu před publikací a ověří, zda mu systém připomíná, aby:

- a. přidal taxonomii.
 - b. přidal náhledový obrázek.
10. Uživatel přidá online kurz.
 11. Uživatel stanoví cenu online kurzu.

Testovací data: pro ověření funkčnosti budou připraveny obrázky, videa a ukázkový text. Také bude existovat účet administrátora.

7.2 Testovací scénáře – uživatelská část

7.2.1 Scénář U1

Cíl: kontrola různých typů přihlášení.

Průchody: uživatel se přihlásí pomocí e-mailu a každé dostupné sociální sítě.

Testovací data: budou registrováni uživatelé, a to prostřednictvím různých sociálních sítí a e-mailu.

7.2.2 Scénář U2

Cíl: prohlédnutí placených foto- a videonávodů bez nákupu členství.

Průchody:

1. Uživatel bez členství otevře placený fotonávod. Ten bude mít některé kroky skryté. Systém zobrazí uživateli hlášku o tom, že návod je placený, a vypíše seznam členství, která může uživatel zakoupit. Objeví se také tlačítko, které uživatele přeměruje k obchodu.

2. Uživatel bez členství otevře placený videonávod. Ten bude zrychlený. Systém zobrazí uživateli hlášku o tom, že návod je placený, a vypíše seznam členství, která může uživatel zakoupit. Objeví se také tlačítko, které uživatele přesměruje k obchodu.

Testovací data: bude vytvořen uživatel bez členství a placené foto- a videonávody.

7.2.3 Scénář U3

Cíl: prohlédnutí placených foto- a videonávodů s členstvím.

Průchody:

1. Uživatel s členstvím otevře placený fotonávod. Ten bude všechny kroky zobrazené. Žádná hláška se neukáže.
2. Uživatel s členstvím otevře placený videonávod. Ten nebude zrychlený. Žádná hláška se neukáže.

Testovací data: bude vytvořen uživatel s členstvím v nějaké kategorii a placené foto- a videonávody ve stejné kategorii.

7.2.4 Scénář U4

Cíl: zápis na online kurz.

Průchody:

1. Uživatel zobrazí online kurz. Bude si moci zobrazit jen náhledové lekce.
2. Uživatel klikne na odkaz pro nákup kurzu a vyplní požadované údaje. Zvolí metodu placení převodem.

3. Administrátor ve svém účtu potvrdí platbu.
4. Uživatel zkontroluje, že může zobrazit libovolnou lekci.

Testovací data: bude vytvořen online kurz a uživatel, který na něj není přihlášen.

7.2.5 Scénář U5

Cíl: přidání příspěvků uživatelem.

Průchody:

1. Uživatel ze svého profilu klikne na odkaz pro přidání fotonávodu.
2. Uživatel přidá fotonávod.
3. Uživatel zkontroluje, že dostal notifikaci o zpracování fotonávodu.
4. Uživatel ověří, že ve svém profilu vidí u tohoto návodu stav „čeká na schválení“.
5. Administrátor zkontroluje, že mu přišel email o čekajícím fotonávodu.
6. Administrátor ověří, že je návod neplacený.
7. Administrátor fotonávod schválí.
8. Uživatel zkontroluje, že dostal notifikaci o schválení.
9. Uživatel zopakuje všechny předchozí kroky pro přidání videonávodu a obrázku.

Testovací data: bude vytvořen uživatel pro přidání příspěvků.

7.3 Výsledky testování

Během projektu proběhlo několik průběžných UAT testování podle scénářů, popsanych v podkapitolách 7.1 a 7.2. Po ověření funkčností byl zákazník

spokojen s aktuálním fungováním systému. Přibyly několik změnových požadavků, uvedených v kapitole 9.2. Finální UAT testování bude provedeno po úpravě designu aplikace podle podkladů, které poskytne třetí strana.



Kapitola 8

Provoz aplikace

V půlce projektu se zásadně změnil požadavek na jeho výstup. Podle původního zadání měla být aplikace „Joaha“ nasazená do reálného prostředí a připravená k okamžitému provozu. To znamenalo, že součástí projektu bylo zvolení hostingu, na kterém systém poběží. Po změnách, popsanych v kapitole 9.2, požadovaným výstupem jsou zdrojové kódy aplikace, kterou následně další tým vývojářů integruje s jinou webovou stránkou. Tím pádem hledání vhodného hostingu je součástí jiného projektu. Nasazení aplikace se plánuje na konec léta 2019, nejdříve ale po ukončení integraci dvou systémů.

Kapitola 9

Vyhodnocení práce z pohledu projektu

K práci jsem přistupovala projektově, a to pomocí harmonogramu (příloha E), sledování rizik a řízení změn. Samotný projekt se rozdělil do dvou etap:

1. Etapa 1 - analýza a výběr CMS systému (zimní semestr).
2. Etapa 2 - implementace a testování (letní semestr).

Během projektu jsem se řídila harmonogramem, vytvořeným ve formě jednoduchého Ganttova digramu. Je vidět 4 typy úkolů:

1. Oranžové - rodičovské úkoly, primární dělení projektu.
2. Modré - akceptace zákazníkem.
3. Zelené - běžné úkoly, které tvoří výstupy projektu.
4. Bílé - nedokončené úkoly, popsané v kapitole 10.

Tato kapitola je věnovaná projektový přístupu, který pomohl minimalizovat zpoždění projektu a zpřehlednit jeho stav pro všechny strany.

příslušnou prioritu a odhadovanou složitost. Používala jsem stejné systémy prioritizace jak i v katalogu požadavků.

■ 9.2.2 Změna výstupu projektu

Na začátku analýzy měla být výstupem projektu plně funkční aplikace, nasazená do prostředí a otestovaná zákazníkem. Aplikace měla být samostatnou jednotkou bez propojení s jinými webovými stránkami.

V průběhu vývoje se zákazník rozhodl změnit samotný koncept aplikace. Kvůli aktualizaci svých webových stránek se rozhodl dvě aplikace propojit. Od tohoto momentu jsem začala koordinovat své aktivity s dalším týmem programátorů. Kvůli tomu, že druhá aplikace je teď ve fázi analýzy, bylo propojení a nasazení aplikace „Joaha“ posunuto na konec léta 2019.

Po domluvě se zákazníkem se tedy výstup projektu změnil z nasazené aplikace na zdrojové kódy, které druhý tým následně použije k propojení obou stránek.

Riziko	Dopad	Výskyt	Eliminace
Změny specifikace	Způsobuje několik problémů, od zpoždění projektu do nemožnosti jeho ukončení.	Během projektu se vyskytlo několik změn, které jsou detailněji popsány v podkapitole 9.2	Před začátkem vývoje jsem provedla analýzu a zadokumentovala jsem požadavky zákazníka. Ty jsem rozdělila do priorit, což umožnilo posunout implementaci Could Have požadavků a věnovat se důležitějším změnám. Další eliminací je to, že zákazník ví o možnosti zpoždění projektu a preferuje počkat, až budou všechny změny implementovány.
Neočekávaný konec spolupráce se zákazníkem.	Ukončení projektu nebo pokračování v práci bez zákazníka.	Na štěstí se toto riziko nenaplnilo.	Dobrou eliminací je zodpovědná práce z mé strany a neustálý kontakt.
Zpoždění v komunikaci	Zpoždění projektu	Riziko se naplnilo. V době odevzdání práce nemám k dispozici podklady pro úpravu vzhledu.	I přes neustálou komunikaci může riziko vzniknout. Proto nejlepší eliminací může být informování zákazníka o možných zpožděních.
Nedostatečná znalost WordPress z mé strany	Nemožnost ukončit projekt nebo jeho zpoždění	Riziko vzniklo na začátku projektu, kdy šla implementace pomalu. S časem jsem se seznámila s CMS a mohla jsem pokračovat rychleji.	Jedinou eliminací bylo v tomto případě čtení dokumentace a seznámení se s prostředím.

Tabulka 9.1: Rizika projektu

ID	Název	Popis	Priorita	Složitost
CR1	Ukládání pozice videa u online kurzu	Systém ukládá pozici videa online kurzu. Když se k němu zákazník vrátí, systém mu nabídne pokračovat v prohlédnutí.	CH	Medium
CR2	Ukládání pozice videa u videonávodu	Systém ukládá pozici videa videonávodu. Když se k němu zákazník vrátí, systém mu nabídne pokračovat v prohlédnutí.	CH	Medium
CR3	Kontrola před publikací	Před publikací systém uživateli zobrazí připomínky, například, že musí definovat, zda je návod placený, nebo přidat náhledový obrázek.	SH	Medium
CR4	Zobrazení návodu pro vkládání příspěvků uživatelem.	Před přidáním příspěvku systém zobrazí uživateli návod, jakého formátu musí být příspěvek.	SH	Low
CR5	Historie příspěvků	Uživateli se zobrazí historie všech zobrazených příspěvků v jeho profilu.	CH	Medium
CR6	Hashtags	Systém umožní uživateli přidávat hashtagy	CH	Medium

Tabulka 9.2: Katalog změnových požadavků



Kapitola 10

Aktuální stav práce

V době odevzdání práce není projekt ukončen, ale blíží se k finální etapě. Po domluvě se zákazníkem budu s vývojem pokračovat i po odevzdání práce. Zákazník nemá se zpožděním problém z důvodů popsanych v podkapitole 9.2.

Zbývá provést následující aktivity:

- Úprava vzhledu aplikace podle designu, který dodá třetí strana.
- Provedení finálního UAT testování aplikace s hotovým designem.
- Implementace případných změnových požadavků, které mohou vzniknout po finálním UAT testování.
- Finální zaškolení zákazníka (počáteční školení proběhla během průběžných UAT testování).

Kapitola 11

Vyhodnocení úspěšnosti projektu

Podle mého názoru se dá projekt považovat za úspěšný. Byly implementovány všechny MH a SH požadavky v rozsahu stanoveném v analýze. Spolupráce se zákazníkem byla příjemná, žádné konflikty nevznikly. Naplněná rizika vedou pouze ke zpoždění projektu, se kterým zákazník souhlasí.

Úspěšnost projektu prokazuje také zpětná vazba zákazníka, kterou mi poskytla majitelka ateliéru ve formě pracovního posudku. Z její slov:

„Daria svou práci vykonávala velmi svědomitě a zodpovědně, vždy včas dle domluveného termínu. Na schůzky byla pokaždé připravená a ochotně mi (a mým kolegům) vysvětlila vzniklé, či vznikající funkce aplikace. Kdykoliv jsem měla na vývoj aplikace nějaký požadavek, Daria v nejbližší době našla nejvhodnější řešení, přestože se s podobnou situací nikdy předtím nesešla. Málokdo, na koho jsem v pracovní sféře narazila, byl tak proaktivní jako právě paní Daria, což si, myslím, je velmi důležité pro obor softwarového inženýrství. Kladné ohodnocení si zaslouží také pro rychlost její práce. Daria s úkoly dlouho neotálí, je tedy znát, že ji práce v jejím oboru opravdu baví. Jako jediné dva malé nedostatky bych zmínila drobnou jazykovou bariéru, kdy paní Daria není rodilou mluvčí a občasná stydlivost, která však po chvílce schůzky vždy opadla.“

Celý posudek lze najít v příloze F. Za účelem ochrany osobních údajů byly razítko a podpis skryté.

Kapitola 12

Závěr

V rámci projektu byla provedena analýza požadavků zákazníka a prvotní návrh řešení stanoveného problému. Všechny požadavky byly zapsány do specifikace, která slouží pro definici rozsahu budoucího systému. Následně byly požadavky zpracovány do katalogu, kde jsou ve formě, jež se hodila pro sledování vývoje. Byl také vytvořen diagram případů užití, který popisuje omezení pro různé role uživatelů. Zpracovala jsem doménový model, který v daném případě slouží jako slovník pojmů, a diagram procesů, který popisuje nejkomplexnější procesy systému. Nedílnou součástí projektu byly wireframes, které pomáhaly při komunikaci se zákazníkem a při sběru požadavků. Další fází projektu byl výběr WordPressu jako CMS systému, ve kterém byla aplikace vyvíjena. Projekt pokračoval implementací aplikace podle zadání a uživatelským testováním. Výstupem projektu jsou podklady pro integraci s jiným systémem zákazníka, a to:

- Specifikace aplikace.
- Katalog požadavků.
- Use Case diagram.
- Domain model.
- Diagram procesů.
- Návrhy obrazovek – wireframes.
- Zdrojové kódy aplikace
- Dokumentace aplikace a uživatelské manuály.

Cílem bylo provést analýzu, vytvořit návrh a implementovat aplikaci podle požadavků zákazníka. Následně bylo nutné systém otestovat a nasadit do reálného prostředí. Došlo k naplnění většiny cílů práce. Během projektu došlo ke změně zadání. Výstupem jsou tedy zdrojové kódy a veškeré podklady, nikoli nasazená aplikace. V době odevzdání práce není projekt ve finálním stavu kvůli zpoždění dodání designu, ale na něm budu dále pokračovat. Zákazník je s průběhem projektu spokojen, hodnotím ho tedy jako úspěšný.

Po ukončení implementační části projektu můžu říct, že správný výběr nástroje pro vývoj – CMS systému WordPress – ušetřil hodně času a je výsledná aplikace mnohem více škálovatelná a udržovatelná než v případě vlastní implementace. Tato práce může být podkladem pro vývojáře, který se začíná seznamovat s WordPressem. Snažila jsem se popsat základní práci s tímto CMS systémem tak, aby vše bylo pochopitelné i pro neznalého uživatele. Tím pádem čtenář má přehled o tom, jak funguje WordPress, a může rozhodnout o jeho použití ve svých projektech. Pro mě projekt byl přínosem jak z hlediska práce s reálným zákazníkem, tak i zkušenosti s novou technologií. Co se týká aplikace „Joaha“, věřím, že po akceptaci zákazníkem a integraci s jinou webovou stránkou bude plnit svůj účel a podporovat komunikaci ateliéru s jeho klienty.



Literatura

- [1] *Atelier Mozaika* [online]. [cit. 2019-01-02]. Dostupné z: <http://ateliermozaika.cz/>.
- [2] *Best Node.js alternatives to WordPress in 2018* [online]. [cit. 2018-12-15]. Dostupné z: <https://react-etc.net/page/best-node-js-alternatives-to-wordpress-in-2018>.
- [3] *BPMN Specification - Business Process Model and Notation* [online]. 25.08.2018. [cit. 2018-12-17]. Dostupné z: <http://www.bpmn.org/>.
- [4] *Brizy – Page Builder* [online]. [cit. 2019-05-04]. Dostupné z: <https://cs.wordpress.org/plugins/brizy/>.
- [5] *CoursePress Learning Management System* [online]. [cit. 2019-05-04]. Dostupné z: <https://cs.wordpress.org/plugins/coursepress/>.
- [6] *Drupal features / PreviousNext* [online]. 22.11.2018. [cit. 2018-12-01]. Dostupné z: <https://www.previousnext.com.au/solutions/drupal/drupal-features>.
- [7] *Elementor Page Builder* [online]. [cit. 2019-05-04]. Dostupné z: <https://cs.wordpress.org/plugins/elementor/>.
- [8] *Fix / Odebrat Wordpress Aktualizovat Chyba (automatické aktualizace WordPress se nepodařilo dokončit. - Prosím Pokus o aktualizaci znovu teď) - Stealth nastavení* [online]. 2011. [cit. 2019-03-20]. Dostupné z: <https://cs.stealthsettings.com/fix-remove-wordpress-update-error-an-automated-wordpress-update-has-failed-to-complete-please-attempt-the-update-again-now.html>.

- [25] *The new Gutenberg editing experience* [online]. [cit. 2019-05-04]. Dostupné z: <https://wordpress.org/gutenberg/>.
- [26] *UML 2 Use Case Diagramming Guidelines* [online]. 04.08.2018. [cit. 2018-12-17]. Dostupné z: <http://www.agilemodeling.com/style/useCaseDiagram.htm>.
- [27] *Understanding Wireframes! - Business Analyst Training Online* [online]. 26.12.2018. [cit. 2018-12-26]. Dostupné z: <http://myprojectanalysis.com/index.php/business-analysis/item/295-understanding-wireframes>.
- [28] *Usage Statistics and Market Share of Content Management Systems for Websites, December 2018* [online]. [cit. 2018-12-05]. Dostupné z: https://w3techs.com/technologies/overview/content_management/all.
- [29] *WAMP, MAMP and LAMP Stack : Softaculous AMPPS* [online]. [cit. 2019-05-05]. Dostupné z: <https://www.ampps.com/>.
- [30] *Web Server* [online]. 2016. [cit. 2018-12-15]. Dostupné z: <https://www.drupal.org/docs/8/system-requirements/web-server>.
- [31] *What Are the Pros and Cons of Joomla, Wordpress, and Drupal?* [online]. [cit. 2019-01-02]. Dostupné z: <https://www.lifewire.com/pros-and-cons-joomla-wordpress-drupal-756555>.
- [32] *WooCommerce Memberships - WordPress Membership Plugin* [online]. [cit. 2019-05-04]. Dostupné z: <https://woocommerce.com/products/woocommerce-memberships/>.
- [33] *WordPress Disadvantages - Not A Good Website Choice* [online]. 02.12.2018. [cit. 2019-01-02]. Dostupné z: <https://www.tbrookswebdesign.com/blog-wordpress-disadvantages.html>.
- [34] *WordPress/gutenberg* [online]. [cit. 2019-03-20]. Dostupné z: <https://github.com/WordPress/gutenberg/issues>.
- [35] *WordPress « WordPress Codex* [online]. 18.02.2019. [cit. 2019-04-08]. Dostupné z: <https://codex.wordpress.org/WordPress>.
- [36] *WPBeginner - Beginner's Guide for WordPress* [online]. 08.04.2019. [cit. 2019-04-08]. Dostupné z: <https://www.wpbeginner.com/>.
- [37] *WP Courses LMS* [online]. [cit. 2019-05-04]. Dostupné z: <https://cs.wordpress.org/plugins/wp-courses/>.
- [38] *WP-Members Membership Plugin* [online]. [cit. 2019-05-04]. Dostupné z: <https://cs.wordpress.org/plugins/wp-members/>.
- [39] *XAMPP Installers and Downloads for Apache Friends* [online]. 26.04.2019. [cit. 2019-05-05]. Dostupné z: <https://www.apachefriends.org/index.html>.

- [40] BHOWMIK, A. *Top 10 Benefits of using Drupal CMS as your website platform.* / LinkedIn [online]. 04/24/2015. [cit. 2018-12-01]. Dostupné z: <https://www.linkedin.com/pulse/top-10-benefits-using-drupal-cms-your-website-platform-amit-bhowmik/>.
- [41] BRENNAN, K. *A guide to the Business Analysis Body of Knowledge®.* International Institute of business analysis, version 2.0 edition, 2009. ISBN 978-0-9811292-1-1.
- [42] CARR, D. F. *Welcoming WordPress 5.0 And The New Editor* [online]. [cit. 2019-03-20]. Dostupné z: <https://www.forbes.com/sites/davidcarr/2018/12/06/welcoming-wordpress-5-0-and-the-new-editor/#78df5cb94a0f>.
- [43] FOWLER, M. *Destilované UML.* Knihovna programátora. Grada, 3 edition, 2009. ISBN 9788024720623.
- [44] GHOST FOUNDATION. *Ghost Publishing Platform Features* [online]. 2013-2018. [cit. 2018-12-01]. Dostupné z: <https://ghost.org/features/>.
- [45] GHOST.ORG. *The Ghost Ecosystem - Core Concepts* [online]. [cit. 2018-12-01]. Dostupné z: <https://docs.ghost.org/concepts/ecosystem/#ghost-android>.
- [46] KEMPKENS, A. *Joomla! - Content Management System to build websites & apps* [online]. [cit. 2018-12-12]. Dostupné z: <https://www.joomla.org/about-joomla.html>.
- [47] LARMAN, C. *Applying UML and patterns.* Prentice Hall PTR, 3rd ed. edition, 2005. ISBN 0-13-148906-2.
- [48] MARSH, A. *Benefits of Drupal website development over other CMS website builder* [online]. 2017. [cit. 2018-12-01]. Dostupné z: <https://medium.com/@marshasely/benefits-of-drupal-website-development-over-other-cms-website-builder-3a9605162c56>.
- [49] NGUYEN, D. Proč Node.js? In NGUYEN, O. D. (Ed.) *Node.js Okamžitě.* Computer Press, 1. vydání edition, 2016. s. 19–20. ISBN 9788025148204.
- [50] SEVERDIA, R. *Joomla! Benefits & Core Features: multilingual, well supported...* [online]. [cit. 2018-12-01]. Dostupné z: <https://www.joomla.org/core-features>.
- [51] SILVER, B. *BPMN method and style.* Cody-Cassidy Press, 2nd ed. edition, 2011. ISBN 9780982368114.
- [52] WORDPRESS.ORG. *Překlady* [online]. [cit. 2018-12-01]. Dostupné z: <https://cs.wordpress.org/preklady/>.
- [53] WORDPRESS.ORG. *WordPress česky* [online]. [cit. 2018-12-01]. Dostupné z: <https://cs.wordpress.org/>.

- [54] WORDPRESS.ORG. *WordPress Mobile Apps* [online]. [cit. 2018-12-01].
Dostupné z: <https://wordpress.org/mobile/>.



Seznam zkratk

BPMN	Business Process Modeling Notation
CH	Could požadavky podle systému MoSCoW
CMS	Content Management System
CSS	Cascading Style Sheets
FAT	Factory acceptance tests, funkcí testy
IDE	Integrated Development Environment, vývojové prostředí
LMS	Learning Management System
MH	Must požadavky podle systému MoSCoW
MS	Microsoft
OS	Operační systém
SEO	Search Engine Optimization
SH	Should požadavky podle systému MoSCoW
UAT	User acceptance test, uživatelské akceptační testy
UML	Unified Modeling Language
WYSIWYG	What you see is what you get - editor, který uživateli rovnou zobrazuje finální vzhled textu.



Slovník pojmů

Admin Bar Horní část administrátorské plochy, používá se jako navigační menu.

Admin Panel Levá část administrátorské plochy, používá se jako primární navigace po administrátorské části webu.

Akce Typ funkce ve WordPressu, která přijme data a zpracuje je. Nic nevrací.

Child theme Česky „šablona-následník“. Používá se pro úpravu funkcí ve WordPressu bez rizika ztráty úprav při aktualizaci. Využívá všechny soubory rodičovské šablony, dokud některý z nich není přepsán.

Codex Online dokumentace systému WordPress.

Filter Typ funkce ve WordPress, která přijme data, zpracuje je a vrátí.

Gutenberg Vestavený vizuální editor WordPress.

MoSCoW Systém prioritizace požadavků.

Shortcode Česky „zkrácený kód“. Výraz typu *[název]* nebo *[název] obsah [název]*, který je přímým voláním funkce backendu z frontendu.

Slug Slovo nebo několik slov, popisujících příspěvek, které se dají použít v URL. Například v odkazu *www.wordpress.org/post/welcome-post* je slugem „welcome-post“.

Taxonomie Způsob grupování příspěvků, například kategorie.

Wireframes Jeden ze způsobů prototypování aplikace, který představuje návrh obrazovky (nejedná se o design, ale o předpokládané rozložení a přehled funkcí).

WP Core Jádro WordPressu, veškerá vestavená funkcionalita.



Příloha A

Specifikace



A.1 Požadovaná funkcionalita

- SH** 5. U placených fotonávodu se sdílí finální obrázek s vodotiskem a odkaz na návod
- SH** 6. U placených videonávodu se sdílí zrychlená verze videa s vodotiskem a odkaz na návod.
- SH** 7. Ke všem video a foto se automaticky přidává vodotisk
- SH** 8. U online kurzů se sdílí úvodní lekce a odkaz na kurz

4.2 Online kurzy

- MH** 1. **Libovolný uživatel** vidí náhledové lekce online kurzu.
- MH** 2. **Libovolný uživatel** může online kurz komentovat.
- MH** 3. **Libovolný uživatel** vidí popis online kurzu.
- MH** 4. **Libovolný uživatel** vidí osnovu online kurzu.
- MH** 5. **Účastník online kurzu** může koukat na libovolné lekce.
- MH** 6. **Účastník online kurzu** může přeskokovat na libovolnou lekci.
- MH** 7. **Účastník online kurzu** může stahovat podklady pro kurz.
- MH** 8. **Libovolný uživatel, kromě účastníku kurzu,** může online kurz koupit.
- CH** 9. **Účastník online kurzu** ve svém profilu uvidí historii nakoupených kurzů.

4.3 Členství

- MH** 1. **Přihlášený uživatel** může zakoupit členství v libovolné kategorii, nebo ve všech kategoriích najednou (úplné členství).
- SH** 2. Členství je časově omezené.
- CH** 3. **Přihlášený uživatel s členstvím** ve svém profilu uvidí historii svých členství.

4.4 Návody

- MH** 1. **Přihlášený uživatel bez členství a nepřihlášený uživatel** vidí jen zkrácenou verzi placeného fotonávodu.
- MH** 2. **Přihlášený uživatel bez členství a nepřihlášený uživatel** vidí jen zrychlenou verzi placeného videonávodu.
- MH** 3. **Přihlášený uživatel s členstvím** v kategorii návodu vidí celou verzi placeného foto/video návodu
- MH** 4. **Přihlášený uživatel s členstvím** v kategorii návodu může stahovat podklady pro placený návod.
- MH** 5. **Libovolný uživatel** vidí celou verzi neplaceného návodu.
- MH** 6. **Libovolný uživatel** může stahovat podklady pro neplacený návod.
- MH** 7. **Libovolný uživatel** může návod komentovat.
- SH** 8. **Přihlášený uživatel** může přidávat své návody. Návody budou neplacené a budou přidáné na web po kontrole **administrátorem**.
- MH** 9. **Přihlášený uživatel** může přidat návod do oblíbených.
- MH** 10. **Libovolný uživatel** může do komentářů přidat obrázek se svou zkušeností.
- CH** 11. Placené a neplacené návody budou graficky odlišné (na placených bude zámek)
- CH** 12. **Přihlášený uživatel** dostane upozornění o tom, že byl vložen nový návod:
 - a. **Frekvenci** upozornění nastaví ve svém profilu
 - b. **Způsob komunikace** (mail, interně aplikaci) nastaví ve svém profilu

4.5 Možnosti administrátora

- MH** 1. Administrátor může ručně měnit role uživatelů.
- MH** 2. Administrátor může mazat komentáře uživatelů.
- MH** 3. Administrátor může přidávat online kurzy a jednotlivé lekce.
- MH** 4. Administrátor může měnit online kurz.
- MH** 5. Administrátor přidává návody.
- MH** 6. Administrátor může měnit návody (název, obrázky apod.)
- MH** 7. Administrátor definuje, zda je návod placený.
- MH** 8. Administrátor může smazat návod.

■ A.2 Komentované návrhy obrazovok



Příloha B

Katalog požadavků

Katalog požadavků

Obecné			Priorita			Vazby			Detail			
ID	Název	Popis	TYP	Priorita	Složítost	Specifika	Use Ca	Uživatelská role	Stav	Zdroj požadavku	Poznámka	Tagy
REQ1	Zákaz tisku fotonávodů	Celý fotonávod se nedá vytisknout	FUNKČNÍ	MH	LOW	4.1.1			ToDo	Zákazník		ochrana, fotonávod
REQ2	Zákaz tisku videonávodů	Celý videonávod se nedá vytisknout	FUNKČNÍ	MH	LOW	4.1.1			ToDo	Zákazník		ochrana, videonávod
REQ3	Zákaz tisku online kurzu	Online kurz a popis k němu se nedá tisknout	FUNKČNÍ	MH	LOW	4.1.1			ToDo	Zákazník		ochrana, kurz
REQ4	Zákaz pravého tlačítka	Pravé tlačítko je zakázané, aby uživatel nemohl stáhnout obrázek	FUNKČNÍ	MH	LOW	4.1.2			ToDo	Zákazník		ochrana
REQ5	Tisk finálního obrázku fotonávodu	U posledního obrázku fotonávodu je povoleno tisk, aby uživatel mohl pracovat podle papírové verze.	FUNKČNÍ	SH	LOW	4.1.3a	UC33	Všichni	ToDo	Zákazník	Finální obrázek vidí každý uživatel nezávisle na druhu přístupu, takže tisk finálního obrázku nezpůsobí únik dat	ochrana, fotonávod
REQ6	Stažení finálního obrázku fotonávodu	U finálního obrázku fotonávodu je povoleno stažení, aby uživatel mohl pracovat podle papírové verze	FUNKČNÍ	CH	LOW	4.1.3b	UC34	Všichni	ToDo	Zákazník	Finální obrázek vidí každý uživatel nezávisle na druhu přístupu, takže tisk finálního obrázku nezpůsobí únik dat	ochrana, fotonávod
REQ7	Sdílení finálního obrázku fotonávodu	U finálního obrázku fotonávodu je povoleno sdílení, aby uživatel mohl obrázkem zaujmout jiné lidi.	FUNKČNÍ	CH	MEDIUM	4.1.3c	UC35	Všichni	ToDo	Zákazník	Finální obrázek vidí každý uživatel nezávisle na druhu přístupu, takže tisk finálního obrázku nezpůsobí únik dat	ochrana, fotonávod, sdílení, fotonávod
REQ8	Sdílení neplacených fotonávodů	Uživatel může sdílet celý neplacený fotonávod	FUNKČNÍ	SH	MEDIUM	4.1.4	UC36	Všichni	ToDo	Zákazník		ochrana, sdílení, videonávod
REQ9	Sdílení neplacených videonávodů	Uživatel může sdílet celý neplacený videonávod	FUNKČNÍ	SH	MEDIUM	4.1.5	UC37	Všichni	ToDo	Zákazník		ochrana, sdílení, videonávod
REQ10	Sdílení placených fotonávodů	Uživatel může sdílet placený fotonávod, a to tak, že sdílí finální obrázek s vodotiskem a odkaz na návod.	FUNKČNÍ	SH	MEDIUM	4.1.5	UC38	Všichni	ToDo	Zákazník		ochrana, fotonávod, sdílení
REQ11	Sdílení placených videonávodů	Uživatel může sdílet placený videonávod, a to tak, že sdílí zrychlenou verzi videa s vodotiskem a odkaz na návod	FUNKČNÍ	SH	MEDIUM	4.1.6	UC39	Všichni	ToDo	Zákazník		ochrana, videonávod, sdílení
REQ12	Vodotisk	Systém automaticky přidá vodotisk ke všem foto a video	FUNKČNÍ	SH	HIGH	4.1.7			ToDo	Zákazník		ochrana
REQ13	Sdílení online kurzů	Uživatel může sdílet online kurz, a to tak, že sdílí úvodní lekci a odkaz na kurz	FUNKČNÍ	SH	MEDIUM	4.1.8	UC40	Všichni	ToDo	Zákazník	Úvodní lekce je vždycky náhledová, tzn. je zdarma	sdílení, ochrana, kurz
REQ14	Náhledové lekce online kurzu	Systém umožní libovolnému uživateli vidět náhledové lekce online kurzu	FUNKČNÍ	MH	HIGH	4.2.1	UC41	Všichni	ToDo	Zákazník	Náhledová lekce je taková lekce, která je dostupná libovolnému uživateli. Slouží k tomu, aby zaujmout uživatele	kurz
REQ15	Komentování online kurzu	Systém umožní libovolnému uživateli komentovat online kurz	FUNKČNÍ	MH	LOW	4.2.2	UC42	Všichni	ToDo	Zákazník		kurz, komentování
REQ16	Popis online kurzu	Systém umožní libovolnému uživateli vidět popis online kurzu	FUNKČNÍ	MH	LOW	4.2.3	UC43	Všichni	ToDo	Zákazník		kurz
REQ17	Osnova online kurzu	Systém umožní libovolnému uživateli vidět osnovu online kurzu	FUNKČNÍ	MH	LOW	4.2.4		Všichni	ToDo	Zákazník		kurz
REQ18	Osnova online kurzu pro účastníka	Účastník kurzu vidí osnovu kurzu jako odkazy	FUNKČNÍ	MH	MEDIUM		UC28	Účastník kurzu	ToDo	Wireframes		kurz
REQ19	Osnova online kurzu pro neúčastníka	Ten, kdo není účastníkem online kurzu, vidí náhledové lekce jako odkazy, ostatní lekce jako text	FUNKČNÍ	MH	MEDIUM		UC44	Všichni kromě účastníka kurzu	ToDo	Wireframes		kurz
REQ20	Přístup k lekcím online kurzu pro účastníka	Systém umožní účastníkovi kurzu koukat na libovolné lekce online kurzu	FUNKČNÍ	MH	MEDIUM	4.2.5	UC30	Účastník kurzu	ToDo	Zákazník		kurz
REQ21	Navigace po lekcím online kurzu	Systém umožní účastníkovi kurzu přeskakovat na libovolnou lekci online kurzu	FUNKČNÍ	MH	LOW	4.2.6	UC29	Účastník kurzu	ToDo	Zákazník		kurz
REQ22	Označení lekce, která se zrovna přehrává	Systém označí lekci, na kterou uživatel zrovna kouká	FUNKČNÍ	MH	MEDIUM			Všichni	ToDo	Wireframes		kurz
REQ23	Stažení podkladů pro kurz	Systém umožní účastníkovi kurzu stáhnout podklady pro kurz	FUNKČNÍ	MH	MEDIUM	4.2.7	UC31	Účastník kurzu	ToDo	Zákazník		kurz
REQ24	Nákup kurzu	Systém umožní libovolnému přihlášenému uživateli, kromě účastníka kurzu, koupit kurz	FUNKČNÍ	MH	HIGH	4.2.8	UC45	Všichni přihlášené uživatele kromě účastníka kurzu	ToDo	Zákazník	Koupit kurz znamená zaplatit cenu kurzu a dostat přístup ke všem lekcím a ke stažení podkladů	kurz, nákup
REQ25	Historie nakoupených online kurzů	Systém zobrazí účastníkovi kurzu v jeho profilu seznam všech nakoupených kurzů, aby o tom uživatel měl přehled	FUNKČNÍ	CH	LOW	4.2.9	UC32	Účastník kurzu	ToDo	Zákazník		kurz, profil
REQ26	Nákup členství	Přihlášený uživatel může zakoupit členství v libovolné kategorii, aby dostal úplný přístup k placeným návodům v této kategorii	FUNKČNÍ	MH	HIGH	4.3.1	UC46	Přihlášený uživatel	ToDo	Zákazník		členství
REQ27	Úplné členství	Přihlášený uživatel může zakoupit úplné členství, tzn. členství ve všech kategoriích, aby získal úplný přístup k placeným návodům ve všech kategoriích	FUNKČNÍ	MH	HIGH	4.3.1	UC46	Přihlášený uživatel	ToDo	Zákazník		členství
REQ28	Časové omezení členství	Členství je časově omezené	FUNKČNÍ	SH	MEDIUM	4.3.2			ToDo	Zákazník		členství
REQ29	Historie členství	Systém zobrazí přihlášenému uživateli v jeho profilu seznam zakoupených členství, a to včetně doby jejich expirace, aby uživatel měl o tom přehled	FUNKČNÍ	CH	LOW	4.3.3	UC23	Přihlášený uživatel s členstvím	ToDo	Zákazník		členství, profil
REQ30	Zkrácená verze fotonávodu	Systém zobrazí uživateli bez členství jen zkrácenou verzi placeného fotonávodu, a to několik kroků a finální obrázek, aby uživatel věděl, o co se jedná, a chtěl tento návod koupit	FUNKČNÍ	MH	HIGH	4.4.1	UC47	Přihlášený uživatel bez členství a nepřihlášený uživatel	ToDo	Zákazník		členství, fotonávod
REQ31	Zrychlená verze videonávodu	Systém zobrazí uživateli bez členství jen zrychlenou verzi placeného videonávodu, aby uživatel věděl, co je to za návod, a chtěl ho koupit	FUNKČNÍ	MH	HIGH	4.4.2	UC48	Přihlášený uživatel bez členství a nepřihlášený uživatel	ToDo	Zákazník		členství, videonávod

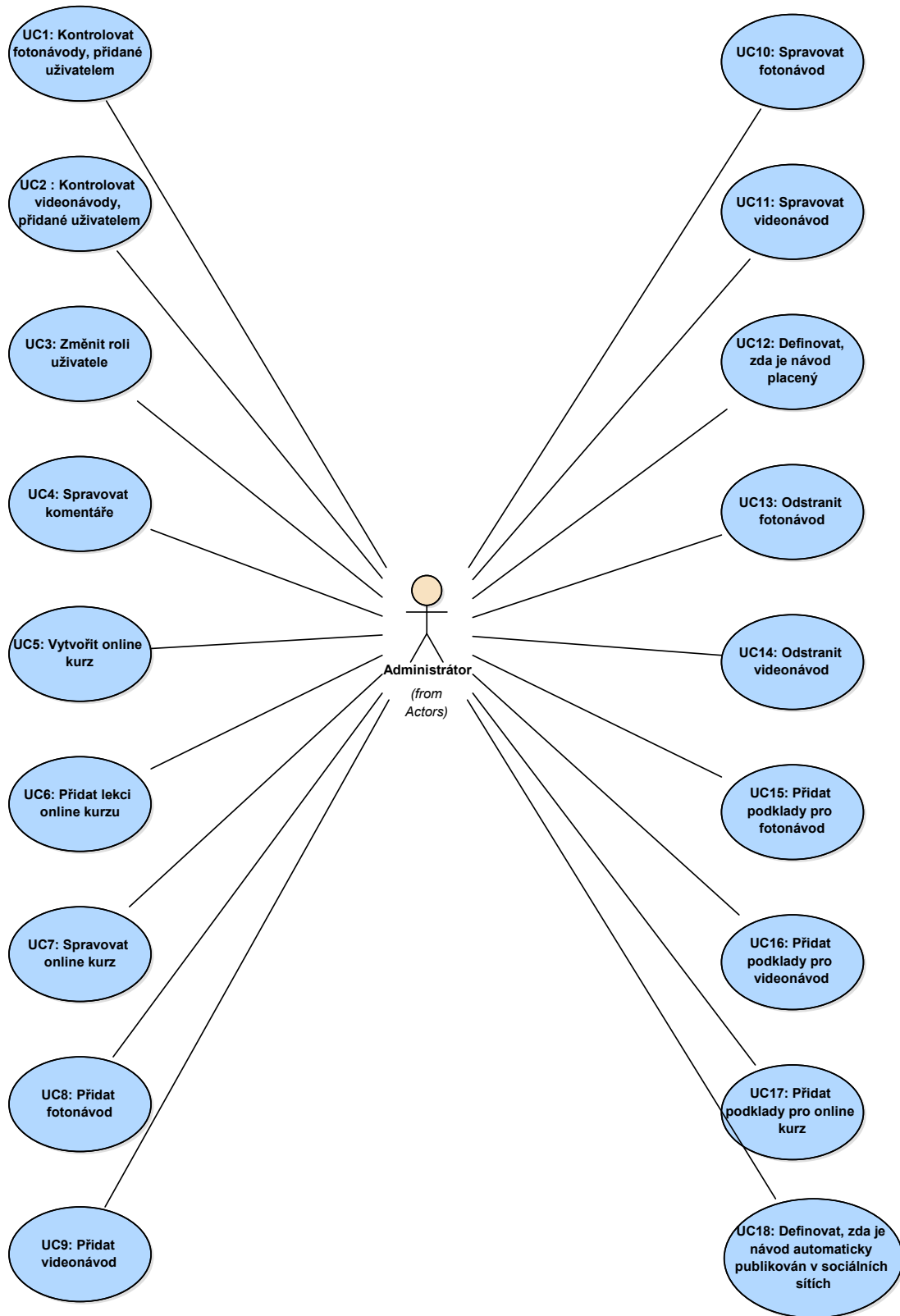


Příloha C

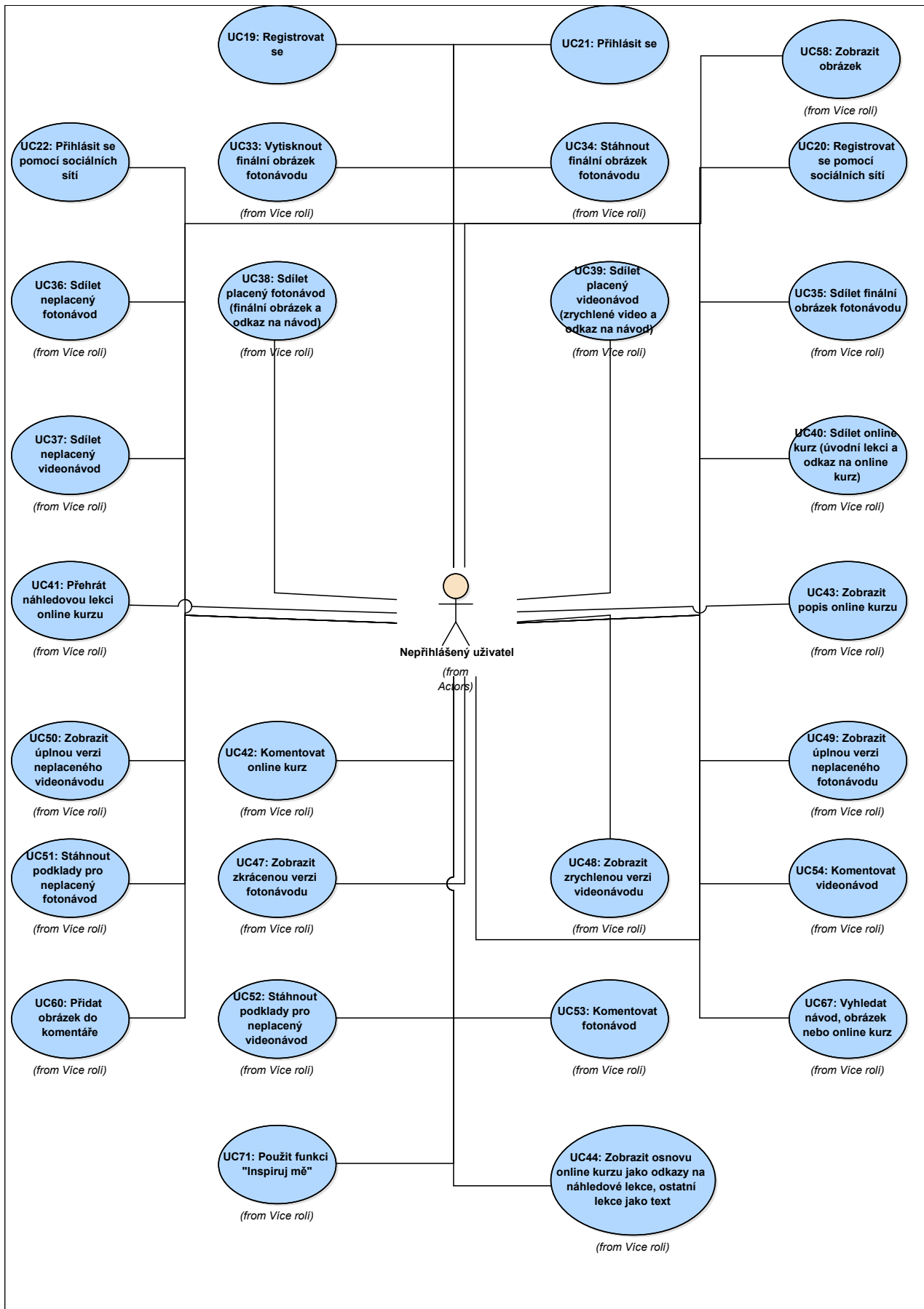
Use case diagram



C.1 Případy užití pro administrátora



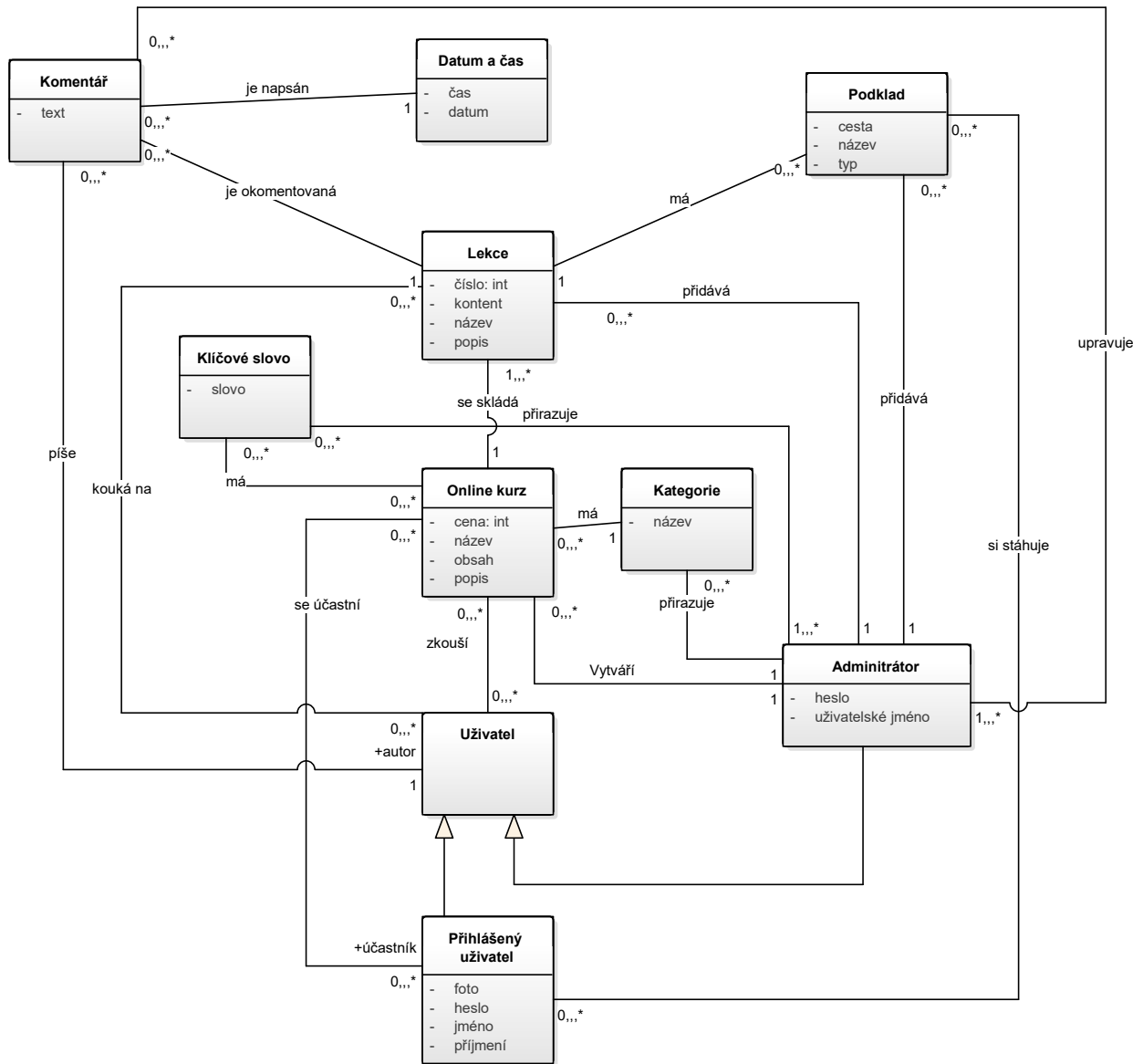
■ C.2 Případy užití pro nepřihlášeného uživatele





Příloha D

Domain model - online kurz





Příloha E

Harmonogram



Příloha F

Pracovní posudek

V Praze dne 3. 5. 2019

Atelier Mozaika s. r. o.

Pracovní posudek

Daria Kulynychenko pracuje pro firmu Atelier Mozaika s. r. o. se sídlem na Praze 6 od 1. 10. 2018 do dnes.

Paní Daria Kulynychenko, studentka softwarového inženýrství na fakultě FEL (univerzita ČVUT) pro mou firmu vytváří projekt v podobě webové aplikace. Hlavní náplní práce paní Daria je analýza a vývoj webové aplikace, která zprostředkovává klientům video a foto návody na kresbu a malbu.

Daria svou práci vykonává velmi svědomitě a zodpovědně, vždy včas dle domluveného termínu. Na schůzky je pokaždé připravená a ochotně mi (a mým kolegům) vysvětluje vzniklé, či vznikající funkce aplikace. Kdykoliv mám na vývoj aplikace nějaký požadavek, Daria v nejbližší době najde nejvhodnější řešení, přestože se s podobnou situací nikdy předtím nesetkala. Málokdo, na koho jsem v pracovní sféře narazila, je tak proaktivní jako právě paní Daria, což si, myslím, je velmi důležité pro obor softwarového inženýrství. Kladné ohodnocení si zaslouží také pro rychlost její práce. Daria s úkoly dlouho neotálí, je tedy znát, že ji práce v jejím oboru opravdu baví. Jako jediné dva malé nedostatky bych zmínila drobnou jazykovou bariéru, kdy paní Daria není rodilou mluvčí a občasná stydlivost, která však po chvilce schůzky vždy opadne.

Celkově musím paní Daria pochválit za její pečlivou, několikaměsíční práci a mohu ji s nejbližším svědomím doporučit dalším klientům.

Pracovní posudek se vydává na žádost studentky ČVUT Daria Kulynychenko.

V Praze dne 3. května 2019

Pavla Macháčková, majitelka Atelieru Mozaika



Příloha G

Struktura CD

CD:.

```
|
+-----analysis
|       | Analyza.EAP
|       | Katalog-pozadavku.xlsx
|       | Secifikace.pdf
|       |
|       \-----wireframes
|
+-----docs
|       +-----html
|       \-----markdown
\-----src
```