

Univerzita Hradec Králové
Fakulta informatiky a managementu
Katedra informatiky

Webová aplikace na platformě Symfony 5
s využitím vlastního cms pro správu obsahu
Bakalářská práce

Autor: Dominik Mach
Studijní obor: Informační management (im3-p)

Vedoucí práce: Mgr. Vojtěch Vorel, Ph.D.

Prohlášení:

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 19.4.2023

.....
Dominik Mach

Poděkování:

Děkuji vedoucímu bakalářské práce Mgr. Vojtěchu Vorlovi, Ph.D. za metodické vedení práce, vstřícný přístup a rychlou odezvu spolu s věcnými připomínkami a radami při kontrolách teoretické i praktické části práce.

Anotace

Cílem bakalářské práce je vytvoření takové webové aplikace, která bude rychlá, přehledná, lehce editovatelná a zároveň podporovat napojení a komunikaci s určitými programy třetí strany, ale také jako koncový bod pro získávání a odesílání potřebných dat prostřednictvím webového rozhraní. Také je plně responzivní a přístupná na většině nejnovějších prohlížečích jak na mobilních zařízeních, tak desktopech. Určená je pro eventové organizace, případně spolky využívající základní webové komponenty a funkce pro komunikaci s API jiných programů. Výsledkem je hotový a komplexní web s tematikou virtuální reality, který je pomocí klíčových slov lehce vyhledatelný v moderních vyhledávačích a zároveň je také velice intuitivní. Aplikace je dobře zabezpečená a splňuje veškeré podmínky moderního webu, ať už se jedná o souhlas s ukládáním dat na počítači uživatele (*cookies*) nebo *GDPR*.

Annotation

Title: Web application built on Symfony framework

The aim of the bachelor thesis is to create a web application that will be fast, clear, easy to edit and at the same time support connection and communication with certain third party programs, but also as an end point for obtaining and sending the necessary data through a web interface. It is also fully responsive and accessible on most of the latest browsers on both mobile devices and desktops. It is intended for event organizations or associations using basic web components and functions for communication with APIs of other programs. The result is a complete and comprehensive website with the topic of virtual reality, which is easily searchable with keywords in modern search engines and is also very intuitive. The application is well secured and meets all the conditions of a modern website, whether it is the consent of cookies or GDPR

Obsah

1	Úvod.....	1
2	Cíl práce.....	2
3	Metodika zpracování	3
3.1	Autorská práva.....	3
4	Popis projektu.....	4
4.1	Popis aplikace	4
4.2	Beat Saber	5
4.3	Využití v praxi.....	5
4.3.1	Kompatibilita	5
	Podpora webových prohlížečů	6
4.3.2	Podpora monitorů a zobrazovacích jednotek.....	6
4.3.3	Testování.....	7
4.4	Responzivita.....	8
4.4.1	Desktopové zobrazení	8
4.4.2	Zobrazení na tabletech	9
4.4.3	Zobrazení na telefonních zařízeních.....	9
5	Struktura.....	10
5.1	Prostředí.....	10
5.2	Využití technologie.....	11
5.2.1	Symfony	11
5.2.2	PHP.....	11
5.2.3	JavaScript.....	12
5.2.4	HTML	12
5.2.5	CSS.....	12
5.2.6	jQuery	13

5.2.7	Twig.....	13
5.2.8	Doctrine.....	13
5.2.9	Bootstrap.....	13
5.2.10	MySQL.....	14
5.2.11	Využití technologie a jejich verze v rámci Symfony.....	14
5.3	Využití návrhových vzorů (design patterns).....	15
5.3.1	Singleton.....	15
5.3.2	Řetězení metod/funkcí.....	15
5.3.3	Znovupoužitelnost.....	15
5.4	Struktura dat aplikace Symfony.....	16
6	UX - Uživatelský zážitek.....	18
6.1	UI - Uživatelské rozhraní.....	18
6.1.1	Vizuální identita.....	19
6.1.2	Formuláře.....	19
6.1.3	Statické stránky.....	20
7	Rozhraní pro správu webu CMS.....	21
7.1.1	Moduly pro správu obsahu.....	22
7.1.2	Grafické rozložení.....	23
7.1.3	Logování změn a akcí.....	23
7.1.4	Testovací účet.....	23
8	Zabezpečení a zálohování.....	24
8.1	Externí zabezpečení.....	24
8.2	Interní zabezpečení.....	24
8.2.1	Autentifikace uživatelů.....	24
8.2.2	Role.....	26
8.3	Git.....	26

8.3.1	Výhody Gitu	26
9	Marketing	27
9.1	SWOT Analýza	27
9.2	SEO	28
9.3	Cookies	28
9.4	Google tag manager.....	29
9.5	Google Analytics	29
9.5.1	Výhody využití Google Analytics	29
9.6	Google reCAPTCHA.....	30
10	Integrace	32
10.1	Synchronizace dat uživatelů.....	32
10.2	Synchronizace dat z externí databáze	32
10.2.1	Volání endpointu	33
10.2.2	Mapování dat.....	33
10.2.3	Popis synchronizace	34
10.2.4	Logování synchronizace.....	34
10.3	Přijímání dat z externích aplikací.....	34
10.3.1	Ukázkový vstup dat z externího zdroje	35
10.4	API Endpoint (export dat).....	35
10.5	Získání tokenu pro autorizaci exportu.....	37
10.6	Export uživatelů.....	38
10.7	Export map	39
10.8	Export výsledků z turnajů	40
10.9	Webhooky	41
10.10	Ostatní propojení.....	41
11	Shrnutí výsledků.....	42

12	Závěry a doporučení.....	43
13	Seznam použité literatury	44
14	Přílohy.....	46

Seznam obrázků

Obrázek 1 - Fotografie úvodní stránky webu	1
Obrázek 2 - Výstřižek z měření rychlosti webu.....	7
Obrázek 3 - Rozložení článku v desktopovém rozlišení.....	8
Obrázek 4 - Mobilní rozlišení.....	9
Obrázek 5 - Tabletové rozlišení.....	9
Obrázek 6 - Využité technologie.....	14
Obrázek 7 - vlastní zpracování, výstřižek dat projektu z aplikace PhpStorm.....	17
Obrázek 8 - Jednotný vizuální styl – výstřižek z aplikace	18
Obrázek 9 - Jednotný vizuální styl - rámečky	21
Obrázek 10 - SWOT analýza aplikace	28
Obrázek 11 - Záznam z měření návštěvnosti stránky.....	30
Obrázek 12 - Požadavky z formulářů	31
Obrázek 13 - Výstup z Instagram API.....	41

Seznam tabulek

Tabulka 1 - Testované prohlížeče.....	6
Tabulka 2 - Kompatibilní parametry zobrazení	6
Tabulka 3 - Využité technologie	11

Seznam ukázek kódu

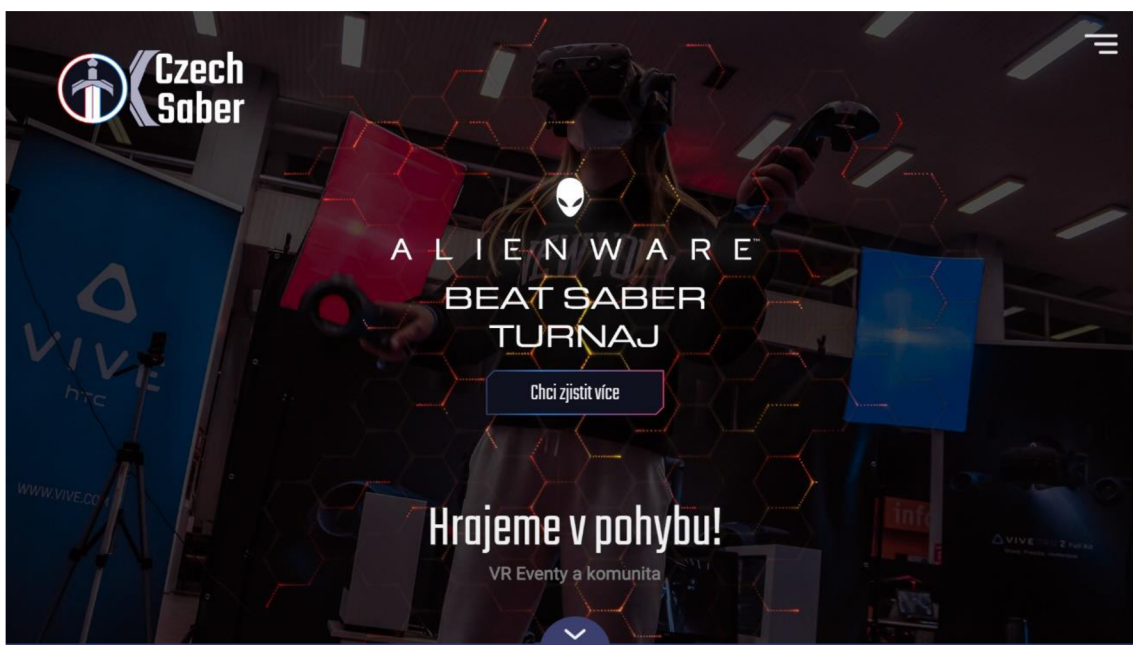
Ukázka kódu 1 - Role a jejich implementace.....	26
Ukázka kódu 2 - Napojení na endpoint.....	33
Ukázka kódu 3 - Mapování dat uživatelů z externí databáze	33
Ukázka kódu 4 - Výstup dat z externího zdroje	35
Ukázka kódu 5 - Metoda pro práci s požadavkem pro export.....	36

Ukázka kódu 6 - Třída pro práci se zpracováním dat z externího API	37
Ukázka kódu 7 - Export dat uživatelů ve formátu json	38
Ukázka kódu 8 - Export dat o mapách ze hry Beat Saber ve formátu json.....	39
Ukázka kódu 9 - Export dat o výsledcích v turnaji ve formátu json.....	40

1 Úvod

Aplikace poskytuje jednoduché a přitom moderní prostředí pro zájemce a fanoušky virtuální reality. Disponuje spoustou funkcí, které ocení každý návštěvník, ať už informovaný o obsahu a funkcích webu, tak i náhodný příchozí. Důležitým prvkem je vlastní administrace webu pro správu obsahu, která je součástí webu a je přehledná a uživatel na první pohled vidí, co může v rámci aplikace editovat, případně kde zjišťovat potřebné informace a data. Obsah stránek je jednoduše editovatelný v rámci této administrace a každý modul má přesně specifikovaný účel. Pro běžného uživatele je výhodou přístup k datům z externích zdrojů, které jsou shromažďovány a poskytovány v českém jazyce a na jednom místě, jako je například aktuální umístění ve světovém žebříčku nebo informace o výkonu jednotlivých uživatelů. Aplikace je všestranně zabezpečená autentizací uživatelů, ochranou proti *SQL Injections* a podchyčením proti skriptům, pomocí kterých by mohla být ohrožena důležitá data.

Pro testovací účely je web spuštěn na adrese: xxiamczechxx.fun



Obrázek 1 - Fotografie úvodní stránky webu

Zdroj: vlastní zpracování

2 Cíl práce

Cílem bakalářské práce je vytvořit takovou webovou aplikaci, která bude splňovat veškeré požadavky moderního webu a bude rozšířená o vlastní moduly pro vytváření turnajů, správu obsahu (*CMS*) a komunikaci s koncovými body (*API*) třetích stran. Jelikož klasické volně dostupné nebo jiné webové editory nenabízí služby typu vytvoření individuální webové stránky pro turnajové eventy, tak je cílem této práce navrhnout a realizovat právě takovou platformu, která bude připravená pro propagaci a eventy ve virtuální realitě a správu obsahu s nastavením viditelnosti ve vyhledávačích. Také bude disponovat funkcemi, které zpříjemní požitky z používání aplikace zejména pro cílovou skupinu uživatelů.

Obsahovat bude několik statických stránek od galerie, novinek až po žebříček a umístění uživatelů. Několika synchronizačními metodami pro aktualizaci dat uživatelů z externích zdrojů a dalších informací, které budou přístupné pro uživatele webu a editovatelné administrátory webu nebo pověřenci, kteří budou mít příslušná práva pro úpravu jednotlivých modulů a obsahu. Zároveň musí splňovat nepsané normy, jako je viditelnost při vyhledávání (*SEO*), o které se starají zejména klíčová slova, meta popisky. Také by měl být intuitivní pro cílovou skupinu a zároveň do jisté míry svojí grafikou drzý, aby upoutal nadšence do virtuální reality a věci spojených s moderními technologiemi na způsob virtuálního světa *Metaverse*.

3 Metodika zpracování

Projekt byl zpracován v programu *PhpStorm* s využitím komponent, jako je terminál, lokální webový server a samotný vestavěný textový editor pro tvorbu webových aplikací. Program jsem vybral na základě přechozích zkušeností. Vzhled webu nevychází z žádných grafických materiálů, veškeré komponenty v uživatelské části byly postupně navrženy a sestaveny při tvorbě webu. Část pro editaci obsahu (*CMS*) je tvořena z volně dostupné (*open-source*) upravené šablony *Bootstrap*.

Web obsahuje propojení s aplikacemi třetích stran pro dosažení toho nejlepšího výsledku. Mezi ně patří například nástroje jako jsou *Google Analytics*, *Google Tag Manager*, *Google ReCAPTCHA*, komunikační platforma *Discord*, externí databáze dat *Scoresaber* a propojení s nástroji *Meta*, využitě pro *Instagram feed*. Na tyto služby se nevztahují žádné omezení, jelikož se jedná o volně dostupné nástroje pro nekomerční účely a jsou určeny zejména pro webové aplikace, aby obohatily jejich funkčnost.

3.1 Autorská práva

Všechny technologie, které aplikace využívá je možné využít jak k nekomerčním, tak komerčním účelům. Na vše, co je v projektu využito, se vztahují základní autorská práva, která nejsou žádným způsobem porušována jejich použitím v bakalářské práci. Veškerý obsah je tvořen mnou samotným. Grafický obsah v podobě obrázků nebo textů využitých ve webové prezentaci mnou samotným nebo někým jiným na základě vzájemné dohody a odsouhlasením s následnou citací autora. Některá data mohou být smyšlená a nemusí přesně vypovídat o názoru nebo správnosti autora textů či obrázků využitých pro tuto práci.

4 Popis projektu

Jedná se o rozsahově menší webovou aplikaci, která má jako cíl oslovit příznivce virtuální reality a zejména hráče hry *Beat Saber*. Největší výhodou je bezkonkurenčnost v Česku a na Slovensku a zároveň specifický design a užitečné funkce, které spojují data ze zahraničních webů s daty hráčů české komunity a nadšenců do virtuální reality. Aplikace je rychlá a poskytuje aktuální informace o dění virtuální reality v regionu ve formě novinek a užitečných informací na jednotlivých stránkách.

Celá aplikace je vyvinuta na platformě *Symfony* 5.4 s použitím externích knihoven a balíčků, které rozšiřují základní funkce tohoto frameworku. Samotný kód je psán v jazyce *PHP*. Jako šablonový framework je využit *twig*, který má bezprostřední kompatibilitu s *PHP* a *Symfony*.

4.1 Popis aplikace

Jedná se o webovou prezentaci pro podporu marketingu, sdílení obsahu a pořádání jednoduchých turnajů s možností zapisování výsledků. Mezi další výhody patří také shromažďování relevantních dat pro české a slovenské uživatele s žebříčkem hráčů. Posloužit může také jen jako marketingový nástroj pro reklamu. Disponuje kontaktním formulářem, modulem pro galerie a novinky, stránkou s informacemi pro aktuálně probíhající turnaj, světovým žebříčkem českých uživatelů aktualizovaný z hlavní a veřejně volně dostupné databáze pro hru *Beat Saber* a v neposlední řadě také slouží jako koncový bod poskytující informace pro ostatní aplikace.

Rozhraní je připraveno zejména pro obsah s virtuální realitou. Některé funkce a moduly je možno využít i na ostatní hry, avšak pro úplnost je potřeba modul programátorem pozměnit nebo přidat nový.

Aplikace není určena přímo ke komerčním účelům jako je prodej produktů nebo poskytování placených služeb.

4.2 Beat Saber

Jedná se o hru ve virtuální realitě, která spadá do žánru rytmických a hudebních videoher. Byla vyvinuta českou společností Beat Games, která se stala známou zejména díky hře *Beat Saber*.

Hráči v této hře hrají se světelnými meči (*sabers*), které drží v ruce a řezají s nimi barevné krychle, které se na ně řítí v rytmu hudby.

Hlavním prvkem, který musí každý hráč mít je zařízení pro zobrazení virtuální reality (*HMD – head-mounted display*), pomocí kterého můžou vnímat svět ve virtuální realitě a umožní jim hrát tuto hru.

Hra umožňuje propojení s největší herní databází (*Scoresaber*) pro tuto danou hru, která obsahuje veškeré výsledky a umístění všech hráčů ve světě. Jedná se o obsáhlou webovou aplikaci v anglickém jazyce, jejímž účelem je shromažďování dat o hráčích a poskytování těchto informací ostatním organizacím tvořící turnaje nebo jiné eventy s virtuální realitou spojené.

4.3 Využití v praxi

Využití vhodné například pro začínající organizaci vyvíjející turnajovou platformu v online prostředí nebo komunitu nadšenců využívající potenciál a možnosti uspořádat jednoduchý turnaj a vše komunikovat prostřednictvím jedné aplikace.

Posloužit může také jen jako online reklamní plocha pro akce a eventy, které jsou s obsahem stránek spojeny. Velikou výhodou je modul pro správu novinek a vkládání galerie s obrázky. Tyto moduly zajišťují dlouhodobou udržitelnost a rozvoj v rámci návštěvnosti stránek.

4.3.1 Kompatibilita

Webová aplikace je kompatibilní s různými typy prohlížečů, operačními systémy a zařízeními, aby byla dostupná pro co nejširší okolí v online světě. To zahrnuje kompatibilitu s nejnovějšími verzemi populárních prohlížečů, jako jsou Google Chrome, Opera GX, Safari a Microsoft Edge, stejně jako s předchozími verzemi těchto prohlížečů.

Kromě toho by měla být webová aplikace kompatibilní s různými operačními systémy, jako jsou Microsoft Windows, MacOS, Linux a také s různými typy zařízení, jako jsou počítače, tablety a chytré telefony.

Podpora webových prohlížečů

Mezi plně podporované prohlížeče patří Safari, Google Chrome, Opera GX a Microsoft Edge a jejich verze znázorněné v tabulce. Ostatní prohlížeče a jejich verze jsou také podporovány, avšak není zaručena 100% kompatibilita zejména u formulářových prvků, responze na mobilních telefonech a starších zobrazovacích jednotkách s nízkým rozlišením.

Tabulka 1 - Testované prohlížeče

Název prohlížeče	Od verze	Do verze
Safari	17613.3.8.0.00	17613.3.9.1.16
Google Chrome	111.0.5563.000	112.0.5615.86
Opera GX	97.0.4719.00	97.0.4719.79
Microsoft Edge	111.0.1661.000	112.0.5615.86

Zdroj: vlastní zpracování na základě testovaných prohlížečů

4.3.2 Podpora monitorů a zobrazovacích jednotek

Podpora je zajištěna pro rozlišení větší jak 1332x776 zobrazovacích bodů pro 16:9 a přiblížení v prohlížeči mezi extrémy 75% až 125%. Při nižším rozlišení nebo jiném přiblížení není zaručené, že budou veškeré prvky zcela správně viditelné a funkční.

Tabulka 2 - Kompatibilní parametry zobrazení

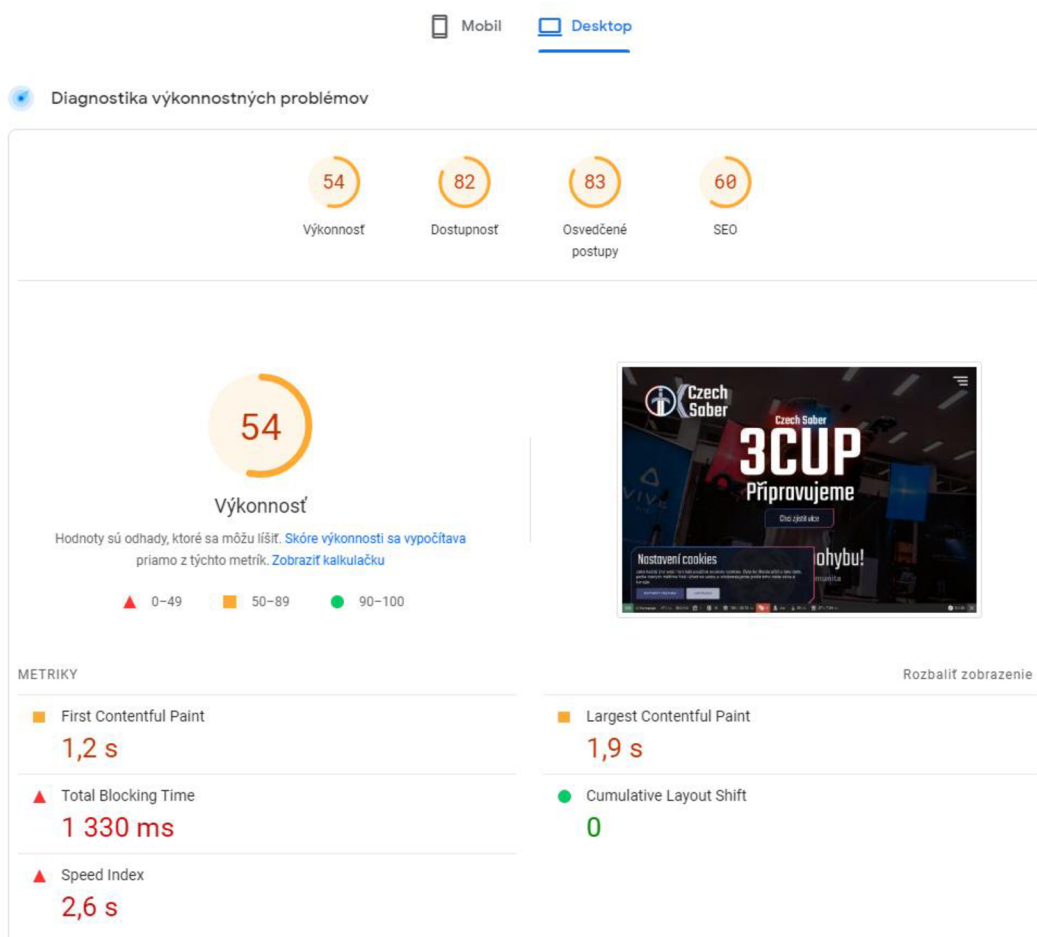
Název zařízení	Rozlišení	Přiblížení
Monitor 21"+	1336x768	75%-125%
Tablet 7"-13"	1024x600	75%-125%
Mobilní telefon 5"-7"	540x960	75%-125%

Zdroj: vlastní zpracování na základě zobrazovacích zařízení

4.3.3 Testování

Prostředí aplikace bylo testováno na mnoha reálných zařízeních v průběhu realizace. Pro testování zobrazení v různých prohlížečích a zařízeních byla využita aplikace *Browserstack*, která nabízí omezené testování zdarma a disponuje funkcí, která dokáže nasimulovat většinu moderních zařízení a prohlížečů a jejich verzí, které je možné si při testování navolit.

Pro testování rychlosti aplikace z hlediska načítání, výkonnosti a dostupnosti byla využita služba *PageSpeed Insights* od společnosti *Google*, která po spuštění otestuje web v desktopovém a mobilním zobrazení a zanalyzuje prvky, které mají negativní vliv na rychlost načítání stránky jako je například využití obrázků s formátem staré generace nebo načtené ale nevyužité CSS styly.



Obrázek 2 - Výstřížek z měření rychlosti webu

Zdroj: vlastní zpracování

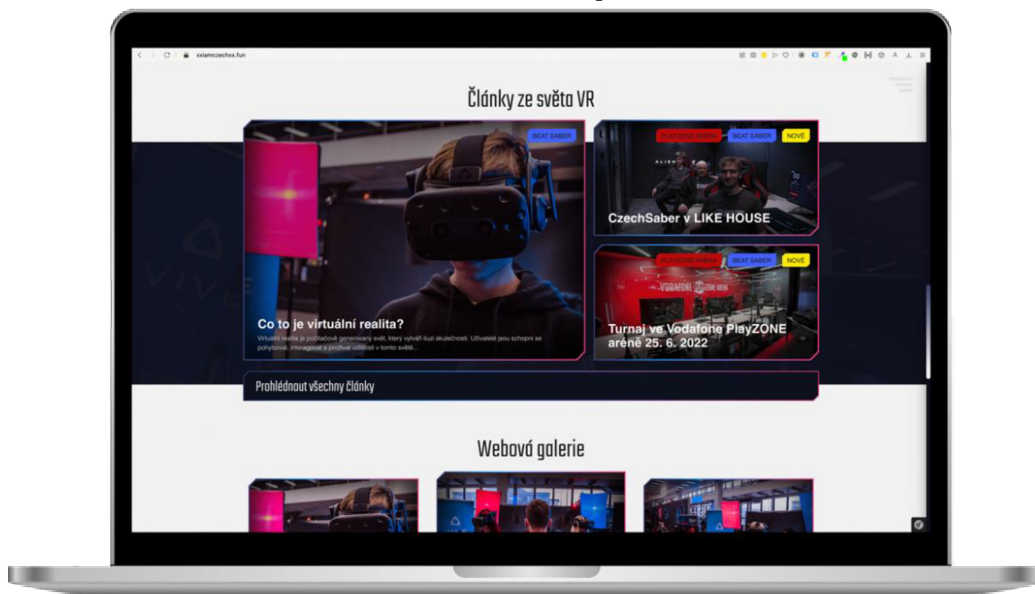
4.4 Responzivita

Řadí se mezi velice důležité prvky moderního webu a je možné ji implementovat pomocí jazyků jako jsou *CSS*, *LESS* nebo *SASS*, které zajistí, že zobrazení stránky bude upraveno pro většinu druhů nejrůznějších zařízení jako jsou mobilní telefony, tablety, notebooky, klasické stolní počítače nebo velkoplošné zobrazovací jednotky. Důležitou funkcí, kterou plní responzivní web je, aby uživatelům procházení webu nesnižovalo koncentraci, motivaci a výkon při procházení stránky na nedesktopovém zařízení. Jedná se o schopnost webové stránky reagovat na velikost obrazovky a orientaci zařízení, na kterém je zobrazena, aby zajistila co nejlepší zobrazení obsahu pro koncového uživatele.

Obsah prvků na stránce je ve většině případů neměnný, pouze se mění způsob jeho zobrazení pro udržení čitelnosti a přehlednosti obsahu. U některých prvků se může jednat i o změnu jejich funkčnosti vzhledem k povaze zařízení, jako je způsob ovládání u dotykových telefonů nebo používání myši na stolním počítači. (Baturay, 2013)

4.4.1 Desktopové zobrazení

Jedná se zatím o nejčastější a nejpoužívanější druh zobrazení, který je specifický svou velikostí a rozlišením. Očekává se využití klávesnice a myši a nejčastěji se jedná o klasické zobrazení na monitorech u stolního počítače nebo notebooku.



Obrázek 3 - Rozložení článku v desktopovém rozlišení

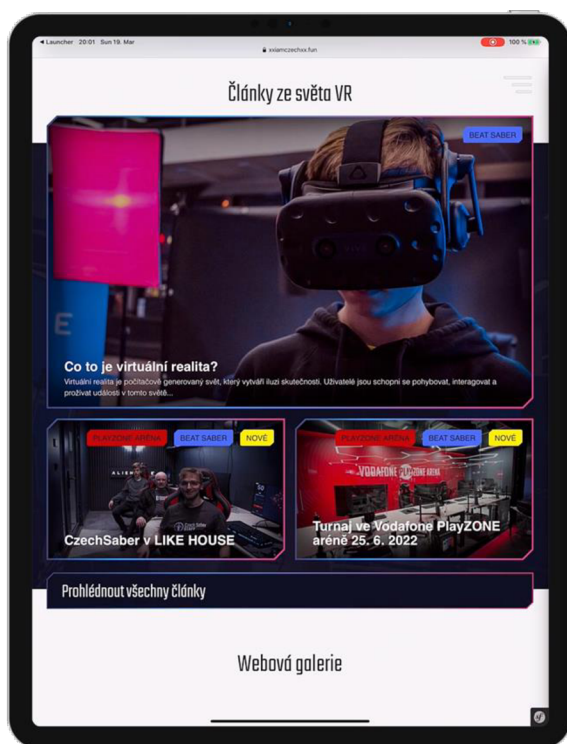
Zdroj: Tento obrázek byl vytvořen s použitím obsahu z <https://www.freepik.com>

4.4.2 Zobrazení na tabletech

Zobrazení, které bývá občas opomíjeno nebo vynecháváno. Vzhledem k stále ještě velkému prostoru, které toto zobrazení nabízí jsou pro tyto poměry upraveny jenom některé komponenty. Očekává se již dotykový přístup interakce s aplikací, proto jsou v tomto zobrazení již k dispozici prvky přizpůsobené pro jednodušší ovládání dotykem, jako jsou tlačítka pro zavírání modálních oken a upravené menu.

4.4.3 Zobrazení na telefonních zařízeních

Mobilní zařízení se stávají čím dál tím více používaným zařízením pro prohlížení webů a aplikací, a proto je aplikace navržena tak, aby se na mobilním zařízení dal obsah stránek stejně tak dobře používat a upravovat, jako je tomu u desktopového zobrazení. Pro toto zobrazení bylo navrženo speciální zobrazení, aby se veškeré prvky vešly i na ta nejmenší zařízení a přitom tak neztratily svoji kvalitu a čitelnost. Jedná se například o zobrazení fotografií v galerii pod sebou nebo přizpůsobení tabulek.



Obrázek 5 - Tabletové rozlišení



Obrázek 4 - Mobilní rozlišení

Zdroj: Tento obrázek byl vytvořen s použitím obsahu z <https://www.freepik.com>

5 Struktura

V této kapitole bude popsáno čím vším v rámci struktury aplikace disponuje od obsahu aplikace z hlediska zdrojového kódu, až po popis jednotlivých bloků, technologií, jazyků a využitých návrhových vzorů pro tvorbu webové aplikace. Do struktury také patří data reprezentována prostřednictvím entit, které jsou obrazem tabulek v databázi, a proto každá entita určuje chování (datový typ) a hodnotu dat náležících dané tabulce. Mezi ně řadíme například tabulku uživatelů, novinek, galerie a turnajů.

5.1 Prostředí

Aplikaci tvořená v prostředí frameworku *Symfony* se dělí na tři základní prostředí, které specifikují chování aplikace a mnohdy také implementace na specifickém serveru.

1. Produkční (*prod – production*) – aplikace je umístěna na veřejně přístupném serveru. Má rozdílný způsob ukládání dočasných souborů (*cache*) a také již není k dispozici lišta, která specifikuje informace o aplikaci.
2. Vývojové (*dev – development*) – prostředí takové, ve kterém se aplikace ve většině případů ještě stále vyvíjí na lokálním stroji. Nemusí tomu tak být ve všech případech. Za podmínky, že není aplikace propagována a přesto už je spuštěna na veřejně přístupném serveru, tak pro odladění chyb a rychlejší práci s aplikací a vývojovou lištou je lepší si ponechat toto prostředí.
3. Testovací (*test – testing*) – prolínají se funkce obou předchozích prostředí. Z vývojového prostředí si přebírá lištu s informacemi o chybách, rychlosti načítání a z produkčního způsob, jakým se ukládají dočasné soubory, aby se dala nasimulovat stabilita aplikace po přechodu na produkční prostředí.

(Potencier, 2007)

5.2 Využité technologie

Veškeré technologie využívané v aplikaci byly v průběhu vypracování postupně aktualizovány na novější verze. Tudiž mohly vzniknout nebo zůstat fragmenty kódu, které jsou naprogramované na starších verzích.

Tabulka 3 - Využité technologie

Název	Typ	Verze	Hrubý odhad využití
Symfony	Framework	5.4	30%
PHP	Jazyk	8.0	32%
JavaScript	Jazyk	1.8.5	1%
jQuery	Knihovna	3.6.4	7%
Twig	Šablonový engine	3.0	12%
Doctrine	ORM Framework	2.9	1%
Bootstrap	Knihovna	5.0	7%
MySQL	Databázový engine	10.4	1%
Sass	Stylovací jazyk	1.5	9%

Zdroj: vlastní zpracování na základě technologií využitých v projektu

5.2.1 Symfony

Symfony je volně dostupný framework pro vývoj webových aplikací zejména v jazyce *PHP*. Poskytuje sadu nástrojů, balíčků a komponent pro rychlejší a efektivnější tvorbu webu a jeho komponent, jako jsou formuláře, směrování, ukládání lokálních souborů (*cookies*) a další. Má silnou podporu pro architekturu *model-view-controller (MVC)* a umožňuje snadnou integraci s dalšími knihovnami a nástroji. (Bondár, 2019; Potencier, 2007)

5.2.2 PHP

Patří mezi nejpoblárnější volně dostupné jazyky pro tvorbu webových aplikací a zde tvoří obrovskou část celé práce. Jedná se o objektově orientovaný jazyk

s obsáhlou dokumentací a mnoha verzemi. Dá se snadno propojit s databázemi, jako je *MySQL* nebo *MariaDB* a s frameworky, jako je právě *Symfony* nebo konkurenční *Laravel*. (Slar, 2018)

5.2.3 JavaScript

Označovaný zkratkou JS je skriptovací jazyk, který se používá pro vývoj interaktivních webových stránek a aplikací. Umožňuje vytvářet interaktivní prvky, jako jsou tlačítka, formuláře a hlavně animace které jsou využity v aplikaci. Také umožňuje komunikaci se serverem prostřednictvím technologií jako je *AJAX* propojující JavaScript a jazyk PHP. Tím poskytuje možnost dynamicky měnit obsah stránek bez přenačtení karty v prohlížeči. (Laurenčík, 2018)

5.2.4 HTML

HTML (*Hypertext Markup Language*) je značkovací jazyk, který se používá pro tvorbu webových stránek. HTML sestává z řady elementů, které jsou označovány značkami a definují různé prvky na webové stránce, jako jsou texty, obrázky, odkazy nebo formuláře. Tvoří nedílnou součást jakékoliv webové aplikace. HTML se používá spolu s dalšími jazyky jako je CSS pro stylování a JavaScript pro interaktivitu na webových stránkách. Ačkoliv už patří mezi zastaralé jazyky, má své následovníky v podobě šablonových frameworků jako jsou *tpl* nebo *twig*, které dokáží lépe pracovat s dnešními frameworky a poskytují balíček funkcí pro práci daty uvnitř šablony. (Laurenčík, 2018)

5.2.5 CSS

CSS (*Cascading Style Sheets*) je jazyk pro stylování webových stránek. Umožňuje definovat vzhled a formát různých prvků na webové stránce, jako jsou barvy, velikosti písma, rozložení apod. Používá se k oddělení formátování od obsahu. Obsahuje pravidla, která se skládají ze selektorů a deklarací a ty mají určité vlastnosti a hodnoty. To pak blíže specifikuje chování vzhledu daného prvku.

Všechny prvky obsažené na webu jsou stylovány pomocí CSS a jeho novější verze SASS, která obohacuje základní sadu CSS o spoustu dalších funkcí a zároveň podporuje kompilaci stylů v reálném čase a podporu pluginů pro kontrolu validity

CSS kódu. Mezi funkce stylovacího jazyku SASS patří například definování barev jako proměnných, využití (*mixin*) funkcí pro snadnější práci s již definovanými styly a zároveň pro znovupoužitelnost kódu. (Laurenčík, 2018)

5.2.6 jQuery

jQuery je knihovna pro jazyk JavaScript, která poskytuje snadný způsob pro selekci, manipulaci a animaci různých *HTML* elementů na webové stránce. Poskytuje řadu užitečných funkcí a metod, které umožňují snadnou práci s *HTML* a snižují komplexitu a náročnost kódu. Je to jedna z nejpoužívanějších JavaScript knihoven a je často používána pro vývoj webových aplikací a interaktivních webových stránek a v projektu je hojně využíván. (Laurenčík, 2018)

5.2.7 Twig

Twig je takzvaný (*template engine*) pro *PHP*, který se často používá v rámci webových frameworků, jako je právě *Symfony*. Chová se jako klasický *HTML* soubor, avšak umožňuje pokročilou práci s proměnnými, cykly a podmínky a disponuje vlastní syntaxí, která umožňuje oddělit logiku aplikace od zobrazení aplikace a tím udržuje kód čistší a snadněji čitelný. Zároveň poskytuje i možnost využití globálních proměnných, které se načítají v rámci celého projektu, což zjednodušuje práci a zároveň upravitelnost opakujících se prvků. (Potencier, 2007)

5.2.8 Doctrine

Doctrine je open-source *ORM* (*Object-Relational Mapping*) pro *PHP*, který se často používá v rámci webových frameworků, jako je právě *Symfony*. *ORM* je nástroj, který slouží k mapování databázových tabulek na objekty *PHP* a umožňuje pracovat s daty v rámci aplikace bez nutnosti využití *SQL*. (Potencier, 2007)

5.2.9 Bootstrap

Bootstrap je volně dostupný framework pro vývoj webových stránek a aplikací, který poskytuje sadu nástrojů pro rychlejší a efektivnější tvorbu šablon a designu webu. Má k dispozici sadu predefinovaných *CSS* tříd a *JavaScript* komponent pro rychle vytváření běžných prvků, jako jsou navigace, formuláře, tlačítka, tabulky a

další. Využitý je zejména v části *CMS*, kde se pracuje se spoustou dat a tabulek. (Potencier, 2007)

5.2.10 MySQL

SQL (*Structured Query Language*) je jazyk, který se používá pro práci s databázemi. Umožňuje vykonávat operace jako je vyhledávání, vkládání, aktualizace a mazání dat v tabulkách. V projektu prakticky nevyužitý, jeho absenci nahrazuje *Doctrine* a pro případné hromadné změny v rámci databáze je možné využití jazyku *MySQL*. (Slar, 2018)

5.2.11 Využité technologie a jejich verze v rámci Symfony

Přiložený Obrázek 5 je výtažkem z konfiguračního souboru samotné aplikace a obsahuje všechny jazyky a balíčky s jejich verzemi, které aplikace buď aktivně využívá nebo jsou volně k dispozici pro použití a nemusí být nutně využívány.

Hvězdička reprezentuje variabilitu aktuální verze vzhledem k ostatním balíčkům. Logický operátor „nebo“ určuje, že aplikace může pracovat na obou verzích tímto operátorem rozdělených.



```
"php": "7.3.0 | 8.0.0",
"ext-ctype": "*",
"ext-iconv": "*",
"ext-json": "*",
"composer/package-versions-deprecated": "1.11.99.2 | 1.11.99.2",
"doctrine/annotations": "1.0 | 1.13.2",
"doctrine/doctrine-bundle": "2.4 | 2.5.2",
"doctrine/doctrine-migrations-bundle": "3.1 | 3.2.1",
"doctrine/orm": "2.9 | 2.10.3",
"knplabs/knp-time-bundle": "1.16 | 1.17.0",
"phpdocumentor/reflection-docblock": "5.2 | 5.3.0",
"soienta/doctrine-json-functions": "5.0 | 5.0.0",
"sensio/framework-extra-bundle": "6.2 | 6.2.6",
"stof/doctrine-extensions-bundle": "1.6 | 1.7.0",
"symfony/asset": "5.3.* | v5.3.11",
"symfony/console": "5.3.* | v5.3.11",
"symfony/debug-bundle": "5.3 | v5.3.4",
"symfony/dotenv": "5.3.* | v5.3.10",
"symfony/expression-language": "5.3.* | v5.3.11",
"symfony/flex": "1.3.1 | v1.17.6",
"symfony/form": "5.3.* | v5.3.11",
"symfony/framework-bundle": "5.3.* | v5.3.11",
"symfony/http-client": "5.3.* | v5.3.11",
"symfony/intl": "5.3.* | v5.3.11",
"symfony/mailer": "5.3.* | v5.3.9",
"symfony/maker-bundle": "1.33 | v1.36.4",
"symfony/mime": "5.3.* | v5.3.11",
"symfony/monolog-bundle": "3.7 | v3.7.1",
"symfony/notifier": "5.3.* | v5.3.11",
"symfony/process": "5.3.* | v5.3.12",
"symfony/property-access": "5.3.* | v5.3.0",
"symfony/property-info": "5.3.* | v5.3.8",
"symfony/proxy-manager-bridge": "5.3.* | v5.3.11",
"symfony/rate-limiter": "5.3.* | v5.3.11",
"symfony/requirements-checker": "2.0 | v2.0.1",
"symfony/runtime": "5.3.* | v5.3.11",
"symfony/security-bundle": "5.3.* | v5.3.12",
"symfony/serializer": "5.3.* | v5.3.12",
"symfony/stopwatch": "5.3 | v5.3.4",
"symfony/string": "5.3.* | v5.3.10",
"symfony/translation": "5.3.* | v5.3.11",
"symfony/twig-bundle": "5.3 | v5.3.10",
"symfony/validator": "5.3.* | v5.3.12",
"symfony/var-dumper": "5.3 | v5.3.11",
"symfony/web-link": "5.3.* | v5.3.4",
"symfony/web-profiler-bundle": "5.3 | v5.3.8",
"symfony/webpack-encore-bundle": "1.12 | v1.13.2",
"symfony/yaml": "5.3.* | v5.3.11",
"twig/extra-bundle": "2.12|^3.0 | v3.3.4",
"twig/intl-extra": "3.4 | v3.4.0",
"twig/twig": "2.12|^3.0 | v3.3.4
```

Obrázek 6 - Využité technologie

Zdroj: vlastní zpracování, upravený obrázek z konfiguračního souboru Symfony

5.3 Využití návrhových vzorů (design patterns)

Návrhové vzory jsou opakovaně použitelné řešení pro běžné problémy při návrhu aplikace. Existuje mnoho různých návrhových vzorů, které se liší svou účelovostí a použitelností.

Některé z nejznámějších návrhových vzorů jsou taktéž v projektu využity a některé jsou samotnou součástí jádra *Symfony* a služeb na něj navázaných.

(Rodríguez, 2015)

5.3.1 Singleton

Zajistí, že existuje pouze jediná instance dané třídy a poskytuje globální přístup k ní. Jako instanci singletonu se dá považovat informace o objektu, kterým je přihlášený uživatel. Z pravidla se jedná o jednoho uživatele v rámci celé instance aplikace, pokud se nejedná o výjimku jako je například aplikace, kde je zpřístupněno uživateli být současně přihlášen pod více účty najednou. (Rodríguez, 2015; Slar, 2018)

5.3.2 Řetězení metod/funkcí

Chain of Responsibility je vzor, který umožňuje poslat požadavek prostřednictvím řetězce objektů, aniž by se požadavek musel znát konkrétní objekt, který ho má zpracovat.

Tento vzor se často používá v situacích, kdy existuje několik možných objektů, které mohou požadavek zpracovat, a není jasné, který z nich by měl být vybrán. (Rodríguez, 2015; Slar, 2018)

5.3.3 Znovupoužitelnost

Jedná se o návrhový vzor, který je svou podstatou velice důležitý a reprezentuje jeden ze základních pilířů programování. Všeobecně kód a zejména jednotlivé třídy a metody by měly mít specifické chování, na které se dá odkazovat z více míst a můžou plnit pro jednotlivá místa jiný účel. (Rodríguez, 2015; Slar, 2018)

5.4 Struktura dat aplikace Symfony

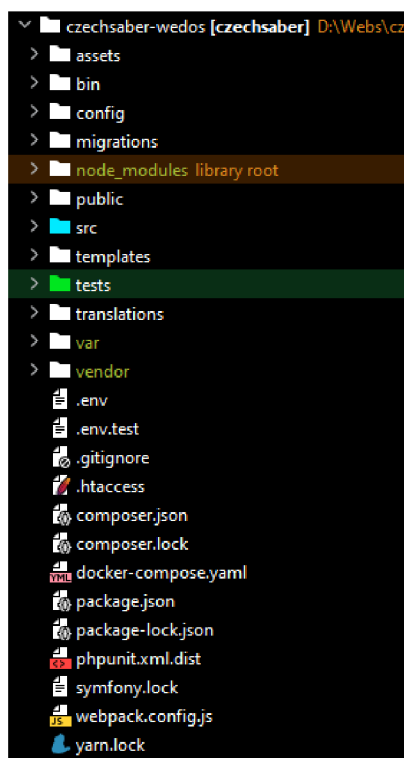
Struktura dat je založena v okamžiku vytvoření projektu. Obsahuje složky, které nesou data samotného frameworku a zároveň složky, které jsou připravené pro data obsahující logiku aplikace. Datová struktura se v různých verzích *Symfony* liší a dá se upravit tak, aby i novější verze pracovaly se soubory stejně jako ty starší a naopak. Například umístění složek s šablonami nebo kontrolerů lze nastavit tak, aby to vyhovovalo jak programátorovi, tak aby aplikace dokázala s jiným seskupením složek bez problémů pracovat. (Fabien Potencier, 2007)

Mezi hlavní složky aplikace patří:

1. Kontrolery (*controllers*) – zodpovídají za logiku aplikace a převod dat mezi modelem a zobrazením (zobrazením koncovému uživateli).
2. Modely (*models*) – představují datový model aplikace a zodpovídají za interakci s databází, také známým jako repositáře.
3. Směrování (*routes*) – určuje, jaké akce se mají provést při zadání určité *URL* adresy. Od *Symfony* 5 již lze routy definovat přímo v *Controllers*, proto jsou zde takřka nevyužité.
4. Šablony (*templates*) – obsahem jsou *twig* šablony s *HTML* kódem, které umožňují zobrazení dat.
5. Složka *config* – složka s konfiguračními soubory aplikace, jako je nastavení databáze.
6. Složka *public* – obsahuje soubory, které jsou přístupné veřejně, jako jsou obrázky, *CSS* a *JavaScript* soubory. Od novějších verzích se o konfiguraci připojení do databází a napojení skriptů starají konfigurační soubory *.env* a *webpack.config.js*
7. Složka *vendor* – disponuje knihovnamy a balíčky, které aplikace využívá.
8. Složka *var* – dočasné soubory (*cache*) pro vývojové i produkční prostředí
9. Složka *tests* – testy aplikace, aby se zajistilo, že aplikace běží správně.
10. Složka *assets* – obsahují veškeré nezkopilované *JavaScript* a *CSS* soubory a zároveň obrázky, které jsou implicitně na webu vloženy programátorem.

11. Soubor yarn.lock – slouží ke kompilaci stylů v reálném čase. Sleduje změny v *SASS* souborech a v *JavaScriptu*.
12. Soubor composer.json – zobrazuje veškeré využití balíčky a pluginy, které webová aplikace využívá a jejich verze.

(Fabien Potencier, 2007)



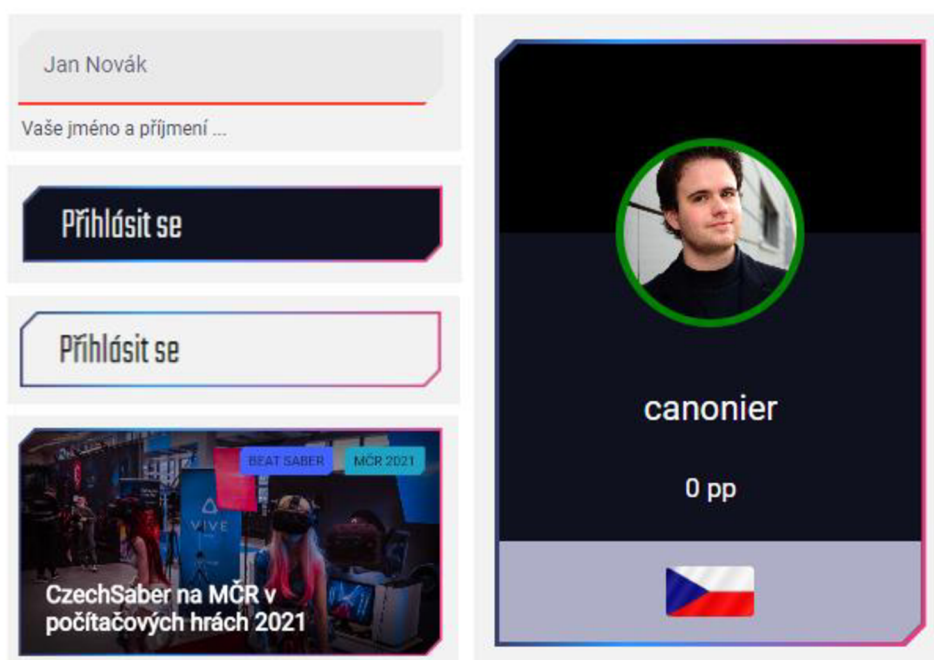
Obrázek 7 - vlastní zpracování, výstřižek dat projektu z aplikace PhpStorm
Zdroj: vlastní zpracování

6 UX - Uživatelský zážitek

UX znamená (*User Experience*), což je v oboru designu a technologie výraz pro způsob, jakým se uživatelé setkávají s webovou aplikací. Jde o interakci mezi uživatelem a aplikací, která se zaměřuje na to, jak efektivně a příjemně uživatel dokáže dosáhnout svého cíle. Cílem UX designu je vytvořit aplikaci, která je intuitivní, efektivní a příjemná pro uživatele. Jelikož se jedná o aplikaci, která má jako účel upoutat příznivce virtuální reality, bylo proto důležité brát na tento faktor velký zřetel a přizpůsobit tomu vzhled a funkčnost celého webu, aby byl na první pohled drzý, ale zároveň přehledný a jednoduchý na manipulaci. (Aizpurua, 2016)

6.1 UI - Uživatelské rozhraní

UI (*User Interface*) je složen z komponent, které slouží jako obsluha uživatele k naplnění cíle co nejjednodušeji a s co největším požitkem. Pro snadnou manipulaci přepínání mezi stránkami je v pravém horním rohu připravený takzvaný *burger*, což je ikona, která po kliknutí vysune pravé menu s tlačítky odkazující na další stránky webu. Patří sem také animace, informační ikony nebo také oznámení, které se objeví po odeslání špatně vyplněného formuláře nebo naopak po úspěšně provedené akci. (Aizpurua, 2016)



Obrázek 8 - Jednotný vizuální styl – výstřižek z aplikace

Zdroj: vlastní zpracování

6.1.1 Vizuální identita

Jednotný vizuální styl je soubor prvků, jako jsou barvy, typografie, ikony, grafické prvky a další, které jsou spojeny s danou značkou, produktem nebo organizací. Tyto prvky se používají konzistentně na všech materiálech, jako jsou právě webové stránky, tiskoviny, reklamní materiály a další.

Cílem jednotného vizuálního stylu je vytvořit jedinečný a soudržný vzhled, který odráží hodnoty a poslání webu a podporuje jeho oblíbenost a zapamatovatelnost. Jedná se zejména o použití prvků na stránkách tak, aby i přes to, že daný uživatel náhodně prochází webem, z něj měl stále stejný pocit. Jako příklad lze uvést roložení takzvaných kontejnerů, které obalují obrázky, tlačítka a nebo texty. Jejich grafické znázornění není stejné, avšak na uživatele působí stejným dojmem, a proto jednotný vizuální styl pomáhá tomu, aby orientace při pohybu po webové stránce byla intuitivní, ale také pro to, aby měl uživatel možnost si tyto prvky zapamatovat a s webem si je i v budoucnu spojit.

6.1.2 Formuláře

Jedná se o prvek, který umožňuje návštěvníkům vyplnit a odeslat určité informace zpět do CMS modulu stránek. Skládají se z prvků, mezi které patří textová pole, tlačítka, rozbalovací nabídky, zaškrtačkové políčka a další prvky, které návštěvníkům umožňují zadávat a odesílat informace prostřednictvím formulářů.

Druhy formulářů pro běžného uživatele:

1. Kontaktní formulář – návštěvníci webové stránky mohou odeslat zprávu nebo dotaz pro majitele stránky nebo pověřence. Tato zpráva se poté objeví v modulu pro odpovědi z formulářů.
2. Registrace formulář – umožňuje uživatelům zaregistrovat se na webové stránce a získat přístup k různým funkcím stránky jako je uživatelský účet nebo s vyšším oprávněním do editoru.
3. Přihlašovací formulář – umožňuje uživatelům přihlásit se na webovou stránku pomocí svého uživatelského jména a hesla.

4. Potvrzení *cookies* – objeví se okamžitě při návštěvě stránky v levém spodním rohu. Obsahuje stručné informace o tom, k čemu se tyto soubory využívají a proč je dobré je potvrdit. Dle nařízení má uživatel právo tento formulář zamítnout a neumožnit tak webu přistupovat k jeho souborům *cookies*.
5. Změna údajů uživatele – tento formulář má k dispozici každý přihlášený uživatel v přehledu účtu a poskytuje mu tak změnu informací tohoto účtu.

6.1.3 Statické stránky

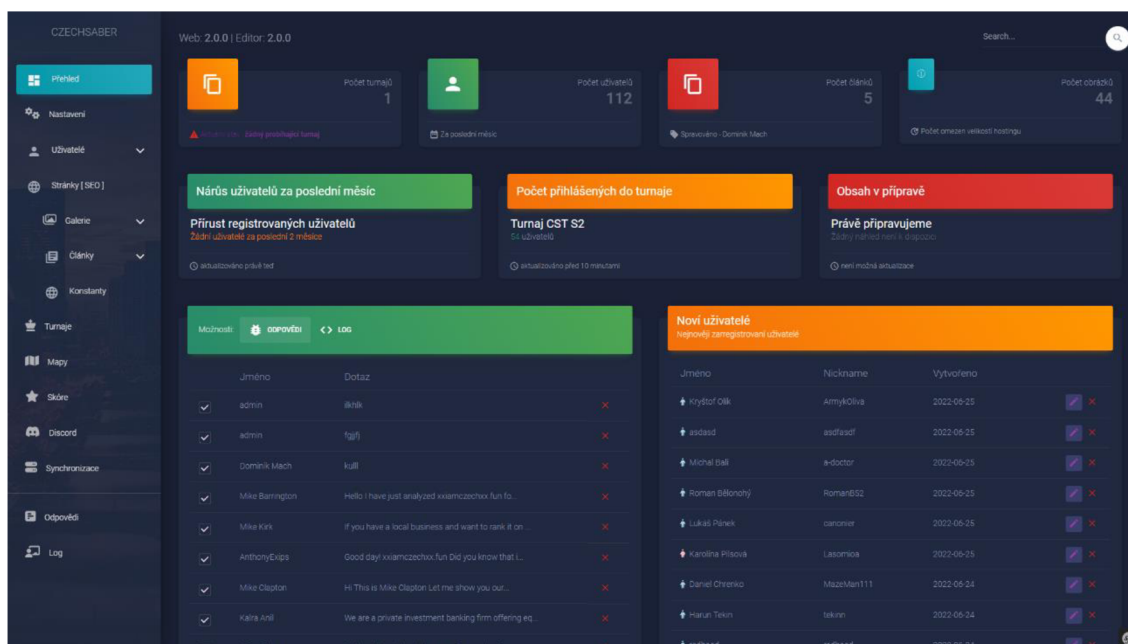
Pro každou stránku je navržený specifický design, který očekává určitý obsah. Jako příklad lze uvést galerii, která obsahuje sekce rozdělené na kategorie s přehledem obrázků. Každá sekce má jednotný vzhled a přihlášený uživatel minimálně s právy pro editaci obsahu stránek může tento obsah editovat a přidávat nebo upravovat jednotlivé obrázky. Statickou stránku lze popsat jako stránku s předurčeným vzhledem a editovatelným obsahem mezi kterou se neřadí například hlavní strana nebo všechny moduly *CMS*. Každá statická stránka má stejnou hlavičku i patičku a mění se pouze obsah a rozložení stránky.

7 Rozhraní pro správu webu CMS

Vlastnoručně vytvořený Content Management System (CMS) slouží k efektivní správě a publikování obsahu na webu. Poskytuje uživatelské rozhraní, které umožňuje uživatelům upravovat a vytvářet obsah na stránce, aniž by potřebovali znát kódování, programování nebo jiné pokročilé technické dovednosti.

Zároveň je přesně navržený pro danou aplikaci a není třeba využití externího systému pro správu obsahu. Obsahuje moduly, ve kterých se nastavují prvky stránky od textů, obrázků, novinek až po nastavení uživatelů, konstant a jiných konfiguračních veličin.

Moduly jsou rozlišeny ikonami a textem v menu a zároveň mají některé moduly specifický vzhled tabulek nebo funkce navíc, jako jsou například export uživatelů nebo odesílání zpráv na *Discord*.



Obrázek 9 - Jednotný vizuální styl - rámečky

Zdroj: vlastní zpracování

Následující jsou obecné funkce, kterými disponuje vlastnoručně vytvořený CMS systém:

1. Správa obsahu umožňuje uživatelům se specifickým oprávněním vkládat, upravovat a mazat obsah na stránkách, jako jsou články, obrázky, videa a další typy obsahu.
2. Správa uživatelů umožňuje administrátorům CMS systému spravovat uživatele a přiřazovat jim role s různými úrovněmi přístupu.
3. Správa šablon umožňuje uživatelům v jisté míře vytvářet a upravovat design a vzhled webových stránek. Jedná se například o rich text editory pro formátování a úpravu textu nebo vkládání obrázků.
4. Správa dat je schopnost CMS systému ukládat data přes konektory stahující informace z jiných webu nebo API koncových bodů.
5. Správa obrázků umožňuje uživatelům nahrávat obrázky, které mohou být použity na stránkách. Jedná se zejména o fotografie použité v galerii webu.
6. SEO optimalizace je zajištěna nástrojem přístupným v každém obsahovém modulu k optimalizaci stránek pro vyhledávače a umožňuje měnit meta popisky, nadpisy a klíčová slova.

7.1.1 Moduly pro správu obsahu

Jednotlivé moduly slouží ke specifickým účelům jako je například správa obsahu stránek, nastavení SEO nebo správa uživatelů. Kromě pár výjimek, má každý modul své tlačítko v navigačním menu a je možné se do něj jednoduše prokliknout. Každý modul má pro svou editaci nebo zobrazení svoje specifická práva. To znamená, že uživatel s menším oprávněním nemusí mít přístup k tomu vidět nebo editovat určitý obsah modulu.

Každý modul obsahuje jednu a více akcí, které může uživatel provést. Jedná se například o editaci pomocí tužičky. Tím je možné editovat například jednotlivé záznamy v tabulce nebo editace celého modulu. Mezi další akce patří přidávání. To slouží k přidávání záznamů a je možné ho nalézt skoro ve všech modulech, kromě výjimek, jako jsou uživatelé. Mezi další akce patří smazání, skrytí/zobrazení

záznamů nebo speciální akce, jako je změna hesla a vygenerování unikátního identivikačního kódu uživatele.

7.1.2 Grafické rozložení

Rozhraní je jednoduché a přehledné. Barvy v tomto prostředí jsou méně výrazné než samotný web a celý modul je ve tmavém režimu (*dark mode*). Na levé straně obrazovky je navigační menu, kterým se lze prokliknout mezi jednotlivými moduly CMS a ve zbylé části je obsah aktuálně navštíveného modulu. Po přihlášení do CMS je k vidění přehled, kde jsou ukázány aktuální statistiky webu, co se počtu uživatelů a novinek týče, ale také aktuální log změn, odpovědí z formulářů na webu a nově zaregistrovaní uživatelé.

7.1.3 Logování změn a akcí

Každá změna spojená s uživatelským účtem v rámci webu je ukládána do logovací tabulky ve které jsou všechny záznamy změn na webu. Mezi ně patří editace obsahu v modulu CMS, ale také například změny nastavení účtu uživatele a potvrzení přihlášení do turnaje na stránce účtu. Nejedná se o měření pohybu uživatele na webu, ani nejsou sbírány žádné informace o návštěvnicích webu, jedná se pouze o změny v databázi způsobené uživatelem, nikoliv samotnou aplikací.

7.1.4 Testovací účet

Velikou výhodou je rozlišování práv uživatelů na základě rolí. To znamená, že uživatelé webu i CMS jsou uloženi v rámci jedné tabulky v databázi. Odlišují se pouze rolí, která poskytuje přístup do CMS rozhraní. Pro testovací účel byl založen uživatel s maximálními právy pod údaji:

- Přihlašovací jméno: admin@admin.cz
- Heslo: admin123

8 Zabezpečení a zálohování

Bezpečné zálohování je nedílnou součástí této aplikace. Vzhledem k výhodám hybridního zálohování, jak serverového (*cloudového*) tak lokálního, je tato metoda využita i zde pomocí záloh ve službě *Git* a lokálních záloh na pevném disku .

8.1 Externí zabezpečení

Jedná se o zabezpečení dat samotné aplikace a záloh.

Jakým způsobem je samotná aplikace zálohována a s jakou úrovní zabezpečení:

1. Záloha aplikace je uložena lokálně na osobním počítači jako zabalený soubor obsahující kompletní implementaci aplikace spolu se zálohou databáze. Zabezpečení je ovlivněno zabezpečením počítače a lokální sítě.
2. Verzování přes *Git*. Jedná se o cloudovou službu, která umožňuje verzování aplikace s vysokou mírou zabezpečení. Zabezpečení je poskytnuto ze strany služby *GitHub*.
3. Složka na serveru ve které je aplikace umístěna. Přístup k datům této složky je ošetřen základní uživatelskou autentizací poskytovanou majitelem hostingové služby. Zabezpečená je jak samotná databáze, tak *FTP* připojení na hosting.

8.2 Interní zabezpečení

V tomto případě se jedná o zabezpečení uživatelských dat na straně webu. Zahrnuje zamezení neoprávněným osobám editovat a upravovat obsah stránek nebo útokům na web formou *SQL Injections*. To jsou takové útoky, pomocí kterých útočník napadne databázi vyplněním formuláře s parametry, které dokáží nad databází webu spouštět dotazy, pomocí kterých mohou uniknout důležitá data.

8.2.1 Autentifikace uživatelů

Framework *Symfony* obsahuje vestavěné nástroje pro autentifikaci uživatelů. Autentizace spočívá v ověření identity uživatele, tedy zda uživatel je oprávněn používat aplikaci, případně její moduly a funkce. Autentizace uživatelů zajištěna

pomocí komponenty Security, která je obsahem balíčku security a ten má vestavěná rozhraní, která jsou ve stručnosti popsána níže v jednotlivých bodech.

Obecný postup využitý pro nastavení autentifikace uživatelů:

1. Konfigurace zabezpečení – určuje se zde zdrojový mechanismus autentifikace a typ autentifikačních providerů, které jsou využívány pro danou autentifikaci. Zároveň obsahuje mimo jiné firewall, přes který lze nasadit omezení počtu nevydařených přihlášení a následné zablokování nebo lze filtrovat požadavky dle IP adresy. To ale v rámci projektu není využito, protože to není žádoucí.
2. Vytvoření uživatele – před autentifikací uživatele je třeba aby byl založen uživatelský účet. Účet je možné založit pomocí registračního formuláře.
3. Implementace uživatelského poskytovatele – uživatelský poskytovatel slouží k poskytnutí informací o uživateli pro autentizační proces. Ve stručnosti ověřuje zadané informace v přihlašovací formuláři a následně zpracovává dle nastavených kritérií. Ověření přihlášení je nastavené na základě hesla a e-mailové adresy, na základě toho hledá informace o uživateli.
4. Implementace autorizačního poskytovatele – autorizační poskytovatel zajišťuje ověření přihlašovacích údajů uživatele.
5. Implementace úspěchu autentizace – pokud autentizace uživatele proběhla úspěšně, je implementována úspěšná odpověď. Tuto odpověď uživatel může zahlédnout jako zeleně podbarvené plovoucí upozornění s informací o úspěšném přihlášení.
6. Implementace chyby autentifikace – když autentizace selže, je implementována chybová odpověď. Tato odpověď se zobrazí jako červeně podbarvené plovoucí upozornění s informací o neúspěšném přihlášení.
7. Zabezpečení kontroly – pro zajištění zabezpečení určitých částí aplikace jsou využity různé metody, například anotace *@Security*.
8. Použití autentifikace – jednotlivé moduly, statické stránky a metody jsou vázané na specifickou roli a povolení se nastavuje v kontrolerech nebo šablonách jednotlivých stránek.

(Fabien Potencier, 2007)

8.2.2 Role

Definují oprávnění uživatelů využívající webovou aplikaci. Každý uživatel může mít jednu nebo více rolí, které mu umožní přistupovat k určitým funkcím nebo omezují jeho přístup k určitým částem aplikace jako je například CMS rozhraní nebo panel pro zápis výsledků.

```
const SUPER_ADMIN = 'ROLE_SUPER_ADMIN'; // Má neomezený přístup
const ADMIN = 'ROLE_ADMIN'; // Má přístup pouze k méně citlivým datům
const COORDINATOR = 'ROLE_COORDINATOR'; // Má přístup pouze do přehledu a turnajům
const EDITOR = 'ROLE_EDITOR'; // Může upravovat novinky, SEO a nastavení webu
const USER = 'ROLE_USER'; // Role přihlášeného uživatele, má přístup do uživ. účtu
```

Ukázka kódu 1 - Role a jejich implementace

Zdroj: vlastní zpracování

8.3 Git

Git je distribuovaný verzovací systém pro správu a sledování změn v kódu. Umožňuje vývojářům uchovávat a spravovat historii svého kódu, spolupracovat na projektech a řešit konflikty při integraci změn od různých lidí. Git je volně dostupný. (Ríos, 2022)

8.3.1 Výhody Gitu

I přes to, že některé z výhod Gitu pro samotného vývojáře nelze na takovém projektu využít, tak nabízí další možnosti, které se mohou pro tvorbu aplikace hodit.

1. Distribuovanost – k dispozici je kompletní historie projektu na svém místním počítači, což znamená, že mohou pokračovat v práci i bez přístupu k dedikovanému serveru.
2. Verzování – umožňuje sledovat a spravovat historii změn v kódu, což umožňuje se jednoduše vrátet ke starším verzím kódu nebo porovnávat rozdíly mezi různými verzemi.
3. Spolupráce – jelikož se jedná o samostatný projekt, není tato výhoda využívána, ale slouží k tomu, že umožňuje více lidem spolupracovat na stejném projektu z různých míst, což zvyšuje efektivitu.
4. Bezpečnost – schopnost vytvářet a udržovat více větví projektu současně a tím vytvářet zálohy kódu a zabraňovat tak ztrátě dat.

(Ríos, 2022)

9 Marketing

Marketing webové aplikace zahrnuje širokou škálu činností, které mají za cíl zvýšit povědomí o aplikaci, přilákat uživatele a zlepšit konverze. Pro úspěšný marketing webové aplikace byly využity následující kroky:

1. Definování cílové skupiny – publikum převážně mužského pohlaví ve věku 12-31 let. Zejména mladiství a peněžně ve střední a vyšší sféře.
2. Vytvoření značky a brandu – aby aplikace reflektovala hodnoty značky jsou využity její rysy i logo.
3. Vytvoření aplikace s prvky podporující marketing – stránky s informacemi, obrázky a novinkami
4. Využití optimalizace pro vyhledavače (*SEO*) – pro zlepšení pozice ve výsledcích vyhledávání. Zajištěno meta popisky, alt popisky a klíčovými slovy.
5. Monitorování a měření výkonu aplikace – využity jsou marketingové nástroje pro měření monitorování pohybu uživatelů na webu jako například Google Analytics.

Marketing webové aplikace je obecně klíčovým faktorem pro úspěch v dnešním digitálním prostředí a vyžaduje důkladnou strategii a pravidelnou úpravu na základě dat a zpětné vazby uživatelů.

9.1 SWOT Analýza

Jedná se o nástroj používaný v oblasti strategického plánování a managementu, který umožňuje analyzovat silné a slabé stránky (*strengths and weaknesses*) a identifikovat příležitosti a hrozby (*opportunities and threats*). (Li-Chen, 2021)

Silné stránky	Slabé stránky
<ul style="list-style-type: none"> - Jednoduchost - Originální design - Speciální funkce - Vlastní CMS 	<ul style="list-style-type: none"> - Nemá potenciál být marketingově soběstačný - Je omezený na region České Republiky a Slovenska
Příležitosti	Hrozby
<ul style="list-style-type: none"> - Jedinečnost v regionu - Komunikace s lidmi prostřednictvím aplikace Discord - Funguje i přes nepříznivé stavy, jako například pandemie 	<ul style="list-style-type: none"> - Zdražování VR zařízení - Příbytek konkurence - Konec podpory hry Beat Saber

Obrázek 10 - SWOT analýza

Zdroj: vlastní zpracování

9.2 SEO

Jedná se o mechanismus optimalizace pro vyhledavače, který obohatí a změní obsah webové stránky za účelem dosažení jejího optimálního umístění na stránkách s výsledky vyhledávačů (*SERP*). Využívá se k zlepšení viditelnosti a pozice stránek v přírodních výsledcích vyhledávačů. Spadá sem správa obsahu, meta popisků a klíčových slov. SEO se zaměřuje na optimalizaci pro konkrétní klíčová slova, která jsou relevantní pro zadané téma nebo oblast. (Wu, 2022; Mavridis, 2023)

9.3 Cookies

Jedná se o malé soubory, které webová stránka ukládá do prohlížeče návštěvníka, aby si stránka mohla zapamatovat některé informace o návštěvníkovi. Cookies se používají pro různé účely, například pro ukládání preferencí uživatele, pro personalizaci obsahu nebo pro sledování chování návštěvníků na stránce pomocí *Google Analytics*.

Mohou obsahovat různé informace, jako například název stránky, ze které návštěvník přišel, nebo informace o preferencích uživatele. Tyto informace jsou uloženy v zakódované formě a nelze je použít k identifikaci konkrétní osoby.

Webové stránky musí dle zákona informovat návštěvníky o používání cookies a získat jejich souhlas s jejich používáním. Pokud uživatel potvrdí tento souhlas, automaticky se spouští pomocí funkcí takzvaných triggerů (*triggers*) služby *Google Analytics* pro měření statistik uživatele a jeho pohybu na stránce. (Koishigawa, 2021)

9.4 Google tag manager

Google Tag Manager (*GTM*) je bezplatný nástroj společnosti Google pro správu a implementaci marketingových a analytických nástrojů na webových stránkách.

Umožňuje vlastníkům webových stránek snadno spravovat a implementovat tagy bez nutnosti přímého přístupu ke zdrojovému kódu stránky. Místo toho mohou vlastníci webových stránek použít rozhraní Google Tag Manager k vložení a spravování tagů přímo z jednoho centrálního bodu.

Nabízí několik nástrojů pro správu a testování tagů, aby se dalo ověřit, že tagy fungují správně, a upravit je, pokud je to nutné. Vhodné zejména při implementaci služeb jako je *Google Analytics*. (Sýkora, 2018)

9.5 Google Analytics

Google Analytics je bezplatný webový analytický nástroj, který umožňuje sledovat a analyzovat návštěvnost a chování návštěvníků na webových stránkách. Pomáhá získat informace o tom, jak návštěvníci používají webovou stránku, včetně demografických údajů, informací o počtu stránek, čase stráveném na jednotlivých stránkách a mnoho dalších. (Sýkora, 2018)

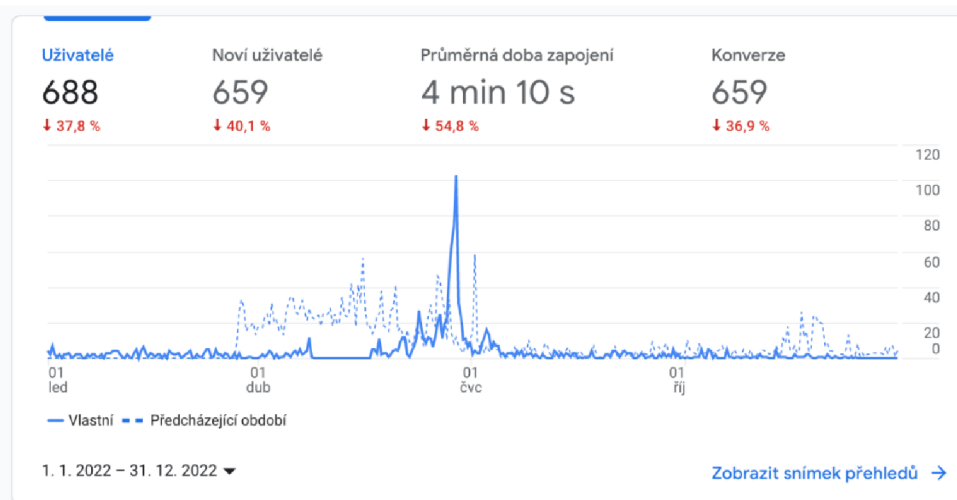
9.5.1 Výhody využití Google Analytics

Ať už pro standartní webovou aplikaci či e-shop mají google analytics řadu výhod.

1. *Google Analytics* je bezplatný nástroj, což umožňuje analyzovat stránku bez vynaložení finančních prostředků.

2. Umožňuje sbírat velké množství informací o návštěvnosti a chování návštěvníků, což umožňuje lepší pochopení jejich potřeb.
3. Může být integrován s dalšími službami *Google*, jako je *Google tag manager*, *AdSense* a *AdWords*.
4. Umožňuje sdílet informace a reporty.
5. Má také k dispozici bezplatnou mobilní aplikaci.

(Sýkora, 2018)



Obrázek 11 - Záznam z měření návštěvnosti stránky

Zdroj: vlastní zpracování

9.6 Google reCAPTCHA

Google reCAPTCHA je bezpečnostní nástroj, který je využit k ochraně před automatickými roboty a spammem. Je to jednoduchý widget, který je implementován na formuláře na webových stránkách a umožňuje určit, zda je návštěvník člověk nebo robot.

Má několik výhod, jako je jednoduchost použití, vysoká účinnost v ochraně před automatickými roboty a spamboty. Tyto výhody pomáhají zajistit bezpečnost a integritu formulářů na webových stránkách a zlepšit uživatelský zážitek.

Nejnovější verze 3 využitá na webu je takzvaná neviditelná (*invisible*) a již neobsahuje widget, ale rozpoznává uživatele na základě jejich pohybu na webových stránkách a pokud jsou přihlášení pod *Google* účtem, tak na základě historie jejich pohybu celkově.

Mezi výhody *Google reCAPTCHA* patří:

1. Ochrana před automatickými roboty a spamem – je navržena tak, aby poskytovala vysokou účinnost v ochraně před automatickými roboty a spamboty.
2. Snadné použití – jednoduchý widget, který může být snadno přidán na formuláře na webových stránkách.
3. Zlepšení uživatelského zážitku – poskytuje jednoduchý a intuitivní způsob, jak ověřit, zda návštěvník je člověk, což má pozitivní vliv na uživatelský zážitek.
4. Více možností verifikace – rozlišuje se zejména verzí a typem. Jako příklad lze uvést vyskakovací okno s obrázky, kde musí uživatel zakliknout specifické obrázky nebo neviditelný rámeček (*widget*), který funguje autonomně bez nutného zásahu uživatele.

(Sýkora, 2018)



Obrázek 12 - Požadavky z formulářů

Zdroj: vlastní zpracování

10 Integrace

Jedná se o propojení webových aplikací a služeb pro dosažení maximálního uživatelského zážitku. Mezi příklady realizací integrací patří RESTful API (*Representational State Transfer*), SOAP (*Simple Object Access Protocol*), nebo GraphQL (*Query Language*). (Wall, 2004)

Příklady webových integracích zahrnutých v aplikaci:

1. Integrace marketingových nástrojů integrace pro komunikaci s návštěvníky webu prostřednictvím propojení (*webhook*) s aplikací *Discord*.
2. Propojení webového rozhraní s externím *API*. Takové rozhraní je využito pro shromažďování dat o uživateli z veřejně přístupné databáze dat s informacemi o hráčích hry *Beat Saber*.
3. Integrace webové aplikace s platformami třetích stran, například sociálními sítěmi, *Google Analytics*, *Google ReCAPTCHA*, apod.

Dobrá webová integrace může také zlepšit uživatelský zážitek tím, že umožní snadnější a rychlejší přístup ke všem potřebným funkcím a informacím.

10.1 Synchronizace dat uživatelů

Prostřednictvím veřejné databáze se dají stahovat informace o hráčích hry *Beat Saber* z celého světa. Jedná se o databázi, která sbírá data pomocí pluginů přímo ze hry a poskytuje tak veškeré informace o hráčích v reálném čase. Výhodou této integrace je, že nemusí uživatelé procházet cizí stránky s žebříčky a složitě vyhledávat to, kde se právě v žebříčku nachází a jaké mají výsledky. Jelikož jsou tyto data synchronizována s jejich účtem na stránkách, je pro ně jednodušší mít tato data pohromadě a porovnávat je přímo s ostatními hráči z České a Slovenské republiky.

10.2 Synchronizace dat z externí databáze

Metoda pro sběr dat je volána buď vstupem uživatele, nebo cronem nastaveným na daný časový interval. Systémový cron je nástroj v operačním systému Unix/Linux, který slouží k plánování a periodickému spouštění automatizovaných úloh (tzv. cron jobů) v určený čas nebo v určitých intervalech. To znamená, že skript je spuštěn automaticky v daný čas a bez nutného zásahu uživatele. (Burlison, 2015)

10.2.1 Volání endpointu

Endpoint je volán *PHP* metodou, která dokáže z endpointu nasbírat data, zkontrolovat jejich validitu a následně přeměrovat dál pro přemapování, uložení do databáze a zobrazení koncovému uživateli.

```
/**
 * @param null $scoresaberId
 * @param string $type
 * @return mixed
 */
public function fetchUserData($scoresaberId = null, string $type = 'full'): mixed
{
    if (null !== $scoresaberId && strlen($scoresaberId) === 17) {
        $context = stream_context_create(array('https' =>
array('header'=>'Connection: close\r\n')));
        try {
            $player =
json_decode(@file_get_contents('https://scoresaber.com/api/player/' . $scoresaberId
. '/' . $type, false, $context), false, 512, JSON_ERROR_NONE);
        } catch (\Exception $e) {
            $player = false;
        }

        return $player;
    }

    return false;
}
```

Ukázka kódu 2 - Napojení na endpoint

Zdroj: vlastní zpracování

10.2.2 Mapování dat

Jedná se o proces, při kterém jsou data z externího API převedena z příslušného kódování a jazyku do podoby, se kterou dokáže web následně pracovat. V tomto kroku je možné s daty pracovat a upravit jejich hodnoty dle potřeby.

```
if ($playerData = $this->fetchUserData($scoresaberId, 'full')) {
    $return = [
        'pp' => (float)$playerData->pp,
        'country' => (string)$playerData->country,
        'rank' => (int)$playerData->rank,
        'countryRank' => (int)$playerData->countryRank,
        'inactive' => (bool)$playerData->inactive,
        'averageRankedAccuracy' => $playerData->scoreStats !== null ?
round((float)$playerData->scoreStats->averageRankedAccuracy, 2) : 0,
        'totalPlayCount' => (int)$playerData->scoreStats?->totalPlayCount,
        'rankedPlayCount' => (int)$playerData->scoreStats?->rankedPlayCount,
        'replaysWatched' => (int)$playerData->scoreStats?->replaysWatched,
        'profilePicture' => (string)$playerData->profilePicture,
        'totalRankedScore' => (int)$playerData->scoreStats?->totalRankedScore,
        'mapped' => 1,
        'level' => $this->calculateLevel((int)$playerData->scoreStats?->
totalRankedScore),
    ];
}
```

Ukázka kódu 3 - Mapování dat uživatelů z externí databáze

Zdroj: vlastní zpracování

10.2.3 Popis synchronizace

Na základě předdefinované url adresy externího endpointu se zavolá požadavek, kterým jsou získány data, které jsou následně namapovány a uloženy na datový server (do databáze). Pro spuštění synchronizace je zapotřebí být přihlášený minimálně jako uživatel s rolí admin.

Dělí se na tři druhy, které plní stejnou funkci, ale jsou spouštěny rozdílně:

1. Synchronizace spouštěná přes CMS – spouští uživatel s příslušnými právy v admin modulu přes tlačítko synchronizace. Nevýhodou je, že při spuštěné synchronizaci je vytížené vlákno, a proto nelze webem procházet.
2. Synchronizace spouštěná cronem – serverový proces (*daemon*) spouští v určitý a předem nastavený čas synchronizační skript. Výhodou je, že nezahltí vlákno uživateli a je ryze autonomní.
3. Synchronizace spouštěná uživatelem webu – ve výpisu uživatelů po rozkliknutí karty uživatele se spustí synchronizační proces, který načte data o uživateli v reálném čase. Výhodou je, že data synchronizována cronem nemusí být v danou chvíli aktuální.

10.2.4 Logování synchronizace

V záložce „Log synchronizace“ jsou informace o posledně běžících synchronizacích, jako je počet synchronizovaných uživatelů, data, která se synchronizovala a úspěšnost.

10.3 Příjímání dat z externích aplikací

Web naslouchá na url adrese <https://xxiamczechxx.fun/api> a zpracovává data, co přijdou z externích zdrojů. Jedná se o zabezpečený import, který očekává na vstupu „secret“. Jedná se o zakódovaný řetězec textu, který dokáže web rozšifrovat a porovnat. Jakmile tento parametr nepřijde nebo přijde a není validní, tak požadavek zahodí.

10.3.1 Ukázkový vstup dat z externího zdroje

```
{
  "secret": "$2a$12$pxKtxVuSdlTyUfsJUE88Bu./ncexiGxsAsGN7t.", (bcrypt hash)
  "mapname": "[Noodle Remaster - OST 1] Jaroslav Beck - Escape", (string)
  "mapid": "a127", (string)
  "players": [
    {
      "name": "Kája", (string)
      "id": "76561198143704479", (integer)
      "score": 46924, (integer)
      "accuracy": 0.166070327162743, (float)
      "combo": 7, (integer)
    },
    {
      "name": "Ema", (string)
      "id": "76561198079969900", (integer)
      "score": 20084, (integer)
      "accuracy": 0.0710799694061279, (float)
      "combo": 2, (integer)
    }
  ],
  "time": "11/27/2021 10:00:43 AM" (datetime)
}
```

Ukázka kódu 4 - Výstup dat z externího zdroje

Zdroj: vlastní zpracování

10.4 API Endpoint (export dat)

API Endpoint je adresa, na které může klient (například webová aplikace) přistupovat k určité funkci nebo datům poskytovaným pomocí rozhraní *API* (*Application Programming Interface*). Endpoint lze tedy chápat jako specifický koncový bod, kde klient může začít komunikovat s *API*. Web vrátí odpověď v jazyce *JSON*,

Každý endpoint v rámci API má svůj vlastní identifikátor, který se skládá z URI (Uniform Resource Identifier), které označuje umístění zdroje, a HTTP metody, která specifikuje, co chce klient s daným zdrojem provést. Například *API*, které umožňuje získávat seznam uživatelů, tak adresa endpointu vypadá: "*https://xxiamczechxx.fun/export?method=users&token=hash*" a *HTTP* metoda musí být typu *GET*. Token je vygenerovaný textový řetězec, který slouží k autentizaci uživatele a bez něj není možné přistupovat k exportům dat.

Může obsahovat také parametry, které klient posílá na server spolu s *HTTP* požadavkem. Tyto parametry mohou být použity k filtrování, řazení nebo omezení počtu výsledků. Endpointy mohou také být zabezpečeny pomocí autentizace a autorizace, aby bylo zajištěno, že pouze oprávnění uživatelé mají přístup k chráněným funkcím a datům. Zde se v rámci exportu nejedná o žádná choulostivá data, a proto je endpoint zabezpečený pouze dle parametru „password“ s hodnotou „admin“. Správná implementace *API* koncových bodů je důležitá pro úspěšnou a efektivní interakci mezi klientem a *API*. (Koci, 2023)

```
/**
 * @Route("/export", name="export")
 */
public function export(Request $request, UsersExport $usersExport, MapsExport
$mapsExport, ScoresExport $scoresExport): Response
{
    $method = $request->query->get('password');
    if ($method !== 'admin') {
        return new JsonResponse(['error' => 'Authentication failed, access
forbidden'], Response::HTTP_FORBIDDEN);
    }

    if ($method = $request->query->get('method')) {
        $data = match ($method) {
            'users' => $usersExport->mapData($this->em),
            'scores' => $scoresExport->mapData($this->em),
            'maps' => $mapsExport->mapData($this->em),
            default => array(sprintf("Unsupported method name %s", $method),
"Supported methods [users, scores, maps]"),
        };
    } else {
        $data = array("No method defined", "In uri slug type parameter
?method=methodName", "Supported methods [users, scores, maps]");
    }

    try {
        return $this->getResponse($data);
    } catch (\JsonException $e) {
        return new JsonResponse(['error' => $e->getMessage()], 500);
    }
}

protected function getResponse($data): Response
{
    $response = new JsonResponse();
    $response->headers->set('Content-Type', 'application/json');

    $response
        ->setContent(htmlspecialchars(json_encode($data, JSON_THROW_ON_ERROR |
JSON_UNESCAPED_UNICODE | JSON_PRETTY_PRINT | JSON_UNESCAPED_SLASHES), 0, 'ISO-8859-
1', true))
        ->setStatusCode(Response::HTTP_OK);

    // $response->headers->set('Content-Disposition', sprintf('attachment;
filename="%s.json"', $user->getUsername())); Downloads response as file attachment
    return $response;
}
```

Ukázka kódu 5 - Metoda pro práci s požadavkem pro export

Zdroj: vlastní zpracování

```

class MapsExport
{
    public function mapData($em): array
    {
        $maps = $em->getRepository(TournamentsMaps::class)->findAll();
        $data = [];

        foreach ($maps as $key => $map) {
            /** @var TournamentsMaps $map */
            $data[$key]['id'] = $map->getId();
            $data[$key]['username'] = $map->getName();
            $data[$key]['bsr'] = $map->getBsr();
            $data[$key]['maxScore'] = $map->getMaxScore();
            $data[$key]['difficulty'] = $map->getDifficulty();
            $data[$key]['pool'] = $map->getPool();
            $data[$key]['inserted'] = $map->getCreatedAt();

            foreach ($map->getPools() as $poolKey => $pool) {
                $data[$key]['availablePools'][$poolKey] = $pool;
            }
        }

        return $data;
    }
}

```

Ukázka kódu 6 - Třída pro práci se zpracováním dat z externího API

Zdroj: vlastní zpracování

10.5 Získání tokenu pro autorizaci exportu

Autorizace exportu je možná pro registrované uživatele, kteří mají povolený export dat. Toto povolení nastavuje správce webu s příslušnými právy a je tak možné mít přehled nad tím, který uživatel má přístup k exportům. Možnost vygenerování nebo získání tokenu je na webové adrese „<https://xxiamczechxx.fun/export-generate-token?username=nickname&password=hash>“. Z hlediska bezpečnosti je pro uživatele v rámci povolení exportu vytvořeno speciální heslo (*hash*), pomocí kterého je možné se dotázat na vygenerování tokenu. Spolu s ním je nutné zadat i přihlašovací jméno (*nickname*) jejich účtu, aby bylo možné ověřit, zda-li se jedná o daný účet.

Tento token je unikátní pro daného uživatele a má neomezenou životnost. Pokud je uživateli následně export zakázán, stává se tím token neplatný a není možné jej použít pro autorizaci.

10.6 Export uživatelů

V rámci exportu dat uživatelů web vrací základní informace o zaregistrovaných uživatelích. Choulostivá data jako hesla, e-mail a další, nejsou z důvodu zachování *GDPR* exportovány.

Webová url adresa i s potřebnou autentifikací pro ověření pomocí vygenerovaného tokenu má podobu: <https://xxiamczechxx.fun/export?method=users&token=hash>

Výsledek z volání požadované url je pole uživatelů, složené z unikátního identifikátoru ID a dat specifikujících daného uživatele:

```
[
  {
    "id": 2,
    "username": "admin",
    "country": "Slovensko",
    "connections": {
      "scoresaberId": 0,
      "twitch": "",
      "discord": "Admin#3526"
    },
    "website": {
      "active": 0,
      "color": "#ffffff"
    },
    "image": "127.0.0.1:8000/uploads/users/e812ccff-0e69-4c25-82a6-7eed932e2980-946.jpg",
    "roles": [
      "ROLE_SUPER_ADMIN",
      "ROLE_ADMIN",
      "ROLE_USER"
    ],
    "scoresaber": {
      "averagePercentage": 0,
      "pp": 0,
      "rank": 0,
      "countryRank": 0
    },
    "created": {
      "date": "2022-01-02 16:57:08.000000",
      "timezone_type": 3,
      "timezone": "UTC"
    }
  },
]
```

Ukázka kódu 7 - Export dat uživatelů ve formátu json

Zdroj: vlastní zpracování

10.7 Export map

Vhodný pro stahování a získávání dat o jednotlivých mapách. Exportovány jsou nejdůležitější informace o mapách, jako je maximální dosažitelné skóre, fáze (*pool*) ve kterém je pro turnaj mapa zařazena spolu s její obtížností. Opět obsahuje unikátní identifikátor ID.

Webová url adresa i s potřebnou autentifikací pro ověření pomocí vygenerovaného tokenu má podobu: <https://xxiamczechxx.fun/export?method=maps&token=hash>

Výsledek z volání požadované url je pole map, složené z unikátního identifikátoru ID a dat specifikujících danou mapu:

```
[
  {
    "id": 3,
    "username": "USAO - Knight Rider",
    "bsr": "1b01b",
    "maxScore": 478515,
    "difficulty": "Normal",
    "pool": "Kvalifikace",
    "inserted": {
      "date": "2022-06-22 23:41:46.000000",
      "timezone_type": 3,
      "timezone": "UTC"
    },
    "availablePools": [
      "Kvalifikace",
      "Minifinále",
      "Semifinále",
      "Gradfinále"
    ]
  },
]
```

Ukázka kódu 8 - Export dat o mapách ze hry Beat Saber ve formátu json

Zdroj: vlastní zpracování

10.8 Export výsledků z turnajů

Jedná se o jednotlivá skóre hráčů odehraných při turnajích. Obsahuje vazbu na mapu a uživatele spolu s dalšími informacemi popisující odehraný pokus.

Webová url adresa i s potřebnou autentifikací pro ověření pomocí vygenerovaného tokenu má podobu: <https://xxiamczechxx.fun/export?method=scores&token=hash>

Výsledek z volání požadované url je pole s výsledky daného pokusu pro určitého uživatele, složené z unikátního identifikátoru ID a dat specifikujících daný výsledek:

```
[
  {
    "id": 7,
    "map": "Forever - Chester Young & STIMX [Chroma]",
    "mapId": 4,
    "tournament": "CST S2 - PlayZONE Arena",
    "score": 247238,
    "userId": 255,
    "userScoresaberId": 0,
    "userNickname": "Malterix",
    "created": {
      "date": "2022-06-25 10:20:40.000000",
      "timezone_type": 3,
      "timezone": "UTC"
    }
  },
]
```

Ukázka kódu 9 - Export dat o výsledcích v turnaji ve formátu json

Zdroj: vlastní zpracování

10.9 Webhooky

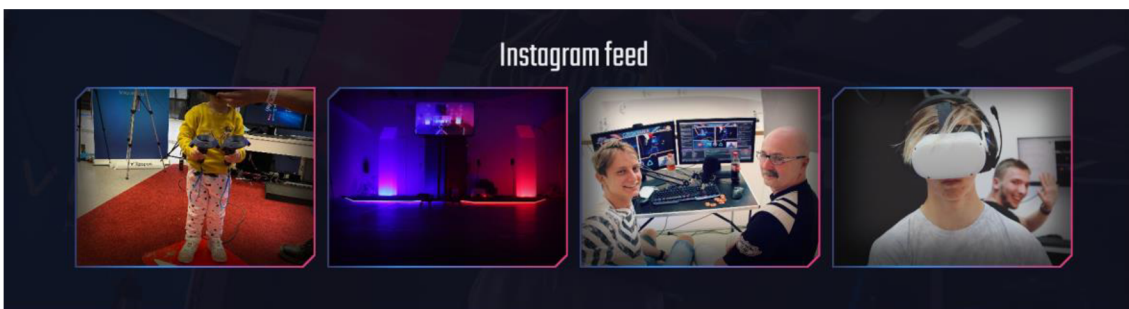
Discord webhook je funkce v aplikaci Discord, která umožňuje externím aplikacím odesílat zprávy do určitého textového kanálu na Discord serveru.

Webhook poskytuje jedinečnou URL adresu, kterou mohou aplikace použít k odeslání zpráv do určeného kanálu. Zpráva může obsahovat textový obsah, odkazy, obrázky, emodži nebo jiné.

Tato forma integrace je využita zejména pro informování uživatelů stránek s propojeným *Discord* účtem nebo hráče turnajů o budoucích změnách nebo eventech. Výhodou této integrace je ta, že zpráva je předem naformátována a administrátorovi stránky pak jen stačí vyplnit drobný formulář, aby odeslal tuto zprávu do předem zvoleného textového kanálu na propojeném *Discord* serveru.

10.10 Ostatní propojení

Na hlavní stránce webu je k dispozici *Instagram feed*, který zobrazuje 4 nejnovější fotky z propojeného instagramového účtu. Tyto fotky jsou prokliknutelné přímo na stránku instagramu a načítají se externě po načtení důležitých prvků webové stránky. Nevýhodou je propojení pomocí klíče, který má omezenou tříměsíční platnost a je třeba ho pravidelně obnovovat. Jedná se o jednoduchou integraci, u které lze nastavit základní parametry zobrazení, jako je například počet fotek.



Obrázek 13 - Výstup z Instagram API

Zdroj: vlastní zpracování

11 Shrnutí výsledků

Aplikaci z hlediska uživatele utvářejí formuláře, statické stránky, obsah a unikátní vizuální identita. Tyto prvky zajišťují prostředí, ve kterém je možné uživateli poskytnout důležitá a relevantní data, které právě potřebuje. Důležitou součástí je i stránka pro zapisování výsledků, která tvoří žebříček s umístěním hráčů v turnaji.

Mimo samotnou webovou aplikaci je k dispozici *CMS* modul pro správu obsahu, který poskytuje grafické prostředí, ve kterém můžou pověřené osoby vytvářet a měnit obsah stránek. Jelikož je součástí aplikace, tak je zcela kompatibilní se samotnou aplikací. Hlavní výhodou je, že je vše implementováno v rámci jednoho rozhraní a není nutnost využití externích *CMS* editorů.

Interní zabezpečení garantuje samostatný framework, který obsahuje balíčky autentifikace a autorizace uživatelů. Důležitou vlastností, kterou aplikace disponuje, je ukládání uživatelů v rámci jedné tabulky v databázi s rozdílnými právy pro přístup do určitých stránek a modulů aplikace. Externí zabezpečení je garantováno klasickým ověřením a zálohováním přes *Git*.

Aplikace dokáže jednoduše pracovat se základními měřícími kódy *Google Analytics* pro měření pohybu uživatelů na webu a konverzí, propojené přes službu *Google Tag Manager*. Tyto měřící kódy jsou spuštěny až po ověření souhlasu uživatelem (*cookies*). Tyto služby zajišťují lepší informovanost o tom, jak k aplikaci uživatelé přistupují a slouží k přímé podpoře marketingu.

Integrace obohacují aplikaci o propojení s externími databázemi a *API*, které poskytují užitečné informace. Tato data jsou v rámci synchronizace stahovány a zpracovávají se pro poskytnutí nejlepšího zážitku koncovému uživateli. Zároveň jsou tato zpracovaná data dále poskytována přes exportní můstek, který je přístupný přes odkaz.

12 Závěry a doporučení

Přes to, že virtuální realita ještě stále není populární ve světě tak jako ostatní technologie, mezi které patří například osobní počítače nebo konzole, tak je zapotřebí tento fenomén podporovat a vytvářet příjemné prostředí pro zájemce o tuto moderní technologii.

Aplikace poskytuje jednoduché a přitom velice moderní prostředí pro zájemce a fanoušky tohoto žánru. Nabízí prostor pro uplatnění na trhu E-sportu a v budoucnu by mohla být žádanou aplikací z hlediska posunu technologií pro virtuální realitu.

Do budoucna by aplikace mohla disponovat monetizací ve formě podpory on-line platebních metod a dalšími funkcemi, které by přispěly k informovanosti a variabilitě pro stávající a nově přibývajícím publikum využívající tuto aplikaci. Práci s pořádáním turnajů by ulehčilo propojení s hrou, které je velice náročné a vyžaduje znalosti jazyka *Python* a zároveň dedikovaný server pro nepřetržitou funkčnost spojení s aplikací.

13 Seznam použité literatury

- [1] Fabien Potencier a François Zaninotto. The Definitive Guide to symfony. 1. edice. Apress, Leden 26, 2007. 455 stran. 978-1590597866.
- [2] David Slar. PHP 7 – Praktický průvodce nejrozšířenějším skriptovacím jazykem pro web. 1. vydání. Zoner Press, 2018. 368 stran. 978-80-7413-363-3.
- [3] Laurenčík Marek. Tvorba www stránek v HTML a CSS. 1. vydání. Grada, 2018. 224 stran. 978-80-271-2241-7.
- [4] Jan Bondár, Úvod do frameworku Symfony: všechno dohromady v harmonii. In: Website [online] 15.3.2019. [cit. 2023-11-017]. Dostupné z: <https://www.root.cz/clanky/uvod-do-frameworku-symfony-vsetko-dohromady-v-harmonii/>
- [5] Kris Koishigawa, Everything You Need to Know About Cookies for Web Development. Freecodecamp [org]. United States, 3.2.2021 [cit. 2023-02-13]. Dostupné z: <https://www.freecodecamp.org/news/everything-you-need-to-know-about-cookies-for-web-development/>
- [6] WU, Ruei-Shian a Huai-Chun LO. A Rose has its Thorn: The role of institutional investors in SEO firms' reporting behavior. *International Review of Economics & Finance*. 2022, **2022**(Volume 82), 530-554. ISSN 1059-0560. Dostupné z: doi:<https://doi.org/10.1016/j.iref.2022.07.004>.
- [7] CHENG, Li-Chen, Kuanchin CHEN, Ming-Chu LEE a Kua-Mai LI. User-Defined SWOT analysis: A change mining perspective on user-generated content. *Information Processing & Management*. 2021, **2021**(Volume 58), 102613. ISSN 0306-4573. Dostupné z: doi:<https://doi.org/10.1016/j.ipm.2021.102613>
- [8] MAVRIDIS, Themistoklis a Andreas L. SYMEONIDIS. Identifying valid search engine ranking factors in a Web 2.0 and Web 3.0 context for building efficient SEO mechanisms. In: SIARRY, Patrick. *Engineering Applications of Artificial Intelligence*. Vytváří se. Francie: IFAC, 2023, s. 75-91. ISBN 0952-1976. 0952-1976. Dostupné z: doi:<https://doi.org/10.1016/j.engappai.2015.02.002>
- [9] BATURAY, Meltem Huri a Murat BIRTANE. Responsive Web Design: A New Type of Design for Web-based Instructional Content. *Procedia - Social and Behavioral Sciences*. 2013, **2013**(Volume 106), 2275-2279. ISSN 1877-0428. Dostupné z: doi:<https://doi.org/10.1016/j.sbspro.2013.12.259>

- [10] RÍOS, Julio César Cortés, Suzanne M. EMBURY a Sukru ERASLAN. A unifying framework for the systematic analysis of Git workflows. *Information and Software Technology*. 2022, **2022**(106811), Volume 145. ISSN 0950-5849. Dostupné z: doi:<https://doi.org/10.1016/j.infsof.2021.106811>
- [11] Free Vectors, Stock Photos & PSD Downloads. *Freepik* [com]. [cit. 2023-04-19]. Dostupné z: <https://www.freepik.com/>
- [12] SÝKORA, Tomáš. Google nástroje, které byste měli vyzkoušet a nic vás nestojí. *Marco B2B* [cz]. Brno: Marco BBN, 2018 [cit. 2023-04-01]. Dostupné z: <https://marcob2b.cz/nastroje-od-googlu-ktere-byste-meli-vyzkouset-a-nic-vas-nejstoji/>
- [13] BURLESON, Donald. Checking and Restarting the Cron Daemon. *Expert Oracle Database Tips by Donald Burleson*. Burleson consulting, 2015, 1. ISSN 9781430262985.
- [14] KOČI, Rediana, Xavier FRANCH, Petar JOVANOVIĆ a Alberto ABELLÓ. Web API evolution patterns: A usage-driven approach. *Journal of Systems and Software*. 2023, 2023(111609). ISSN 0164-1212. Dostupné z: doi:<https://doi.org/10.1016/j.jss.2023.111609>.
- [15] WALL, David. Chapter 4 - Simple Object Access Protocol Under PHP. In: KAUFMANN, Morgan. *Multi-Tier Application Programming with PHP: Multi-Tier Application Programming with PHP*. 2004. 2004, s. 45-54. ISBN 9780127323503.
- [16] AIZPURUA, Amaia, Simon HARPER a Markel VIGO. Exploring the relationship between web accessibility and user experience. *International Journal of Human-Computer Studies*. **2016**(Volume 91), 13-23. ISSN 1071-5819. Dostupné z: doi:<https://doi.org/10.1016/j.ijhcs.2016.03.008>.
- [17] RODRÍGUEZ, Francy D., Silvia T. ACUÑA a Natalia JURISTO. Design and programming patterns for implementing usability functionalities in web applications. *Journal of Systems and Software*. 2015, **2015**(Volume 105), 107-124. ISSN 0164-1212. Dostupné z: doi:<https://doi.org/10.1016/j.jss.2015.04.023>.

14 Přílohy

1) Zadání bakalářské práce



Univerzita Hradec Králové
Fakulta informatiky a managementu

Zadání bakalářské práce

Autor: Dominik Mach

Studium: I2000502

Studijní program: B0688A140001 Informační management

Studijní obor: Informační management

Název bakalářské práce: **Webové stránky na platformě Symfony 5 s vlastní správou obsahu**

Název bakalářské práce Website using the Symfony 5 platform with a custom CMS
AJ:

Cíl, metody, literatura, předpoklady:

Cílem práce je vytvořit kompletní webovou stránku s využitím nejnovějších frameworků pro tvorbu webových aplikací, která obsahuje vlastní systém pro správu obsahu (CMS).

Mezi klíčové využití technologie patří framework pro tvorbu aplikací v jazyce php s názvem Symfony verze 5, framework Doctrine ORM obsahující set knihoven s objektově orientovaným přístupem pro správu databází, skriptovací jazyk sass pro kódování frontendu webu a na animace využita knihovna jQuery pro jazyk javascript.

Výsledkem praktické části bude samostatná webová stránka s možností dynamické editace obsahu stránek, nastavení SEO pro vyhledavače a obsahovat bude také vlastní systém pro správu uživatelů na webu. Mezi klíčové vlastnosti patří propojení s aplikacemi třetích stran jako je Discord WebHook, Twitch a Instagram.

1) The Definitive Guide to symfony

Autoři: Fabien Potencier a François Zaninotto
Rok vydání: 2007

2) PHP 7 - Praktický průvodce nejrozšířenějším skriptovacím jazykem pro web

Autor: RNDr. Jan Pokorný
Rok vydání: 2018

3) Tvorba www stránek v HTML a CSS

Autor: Marek Laurenčík
Rok vydání: 2019

Zadávací pracoviště: Katedra informatiky a kvantitativních metod,
Fakulta informatiky a managementu

Vedoucí práce: Mgr. Vojtěch Vorel, Ph.D.

Datum zadání závěrečné práce: 26.1.2021

