UNIVERZITA KOMENSKÉHO V BRATISLAVE

FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

SYSTÉM NA PODPORU PRÁCE V SKUPINE

BAKALÁRSKA PRÁCA

2021

ANNA REBEKA SOJKA

UNIVERZITA KOMENSKÉHO V BRATISLAVE

FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

SYSTÉM NA PODPORU PRÁCE V SKUPINE

BAKALÁRSKA PRÁCA

Študijný program: Aplikovaná informatika

Študijný odbor: 2511 Aplikovaná informatika

Školiace pracovisko: Katedra didaktiky matematiky, fyziky a informatiky

Školiteľ: doc. RNDr. Ľudmila Jašková, PhD.

2021

ANNA REBEKA SOJKA

ZADANIE

### Čestné prehlásenie

Čestne prehlasujem, že bakalársku prácu Systém na podporu práce v skupine

som vypracovala samostatne s použitím uvedenej literatúry a pod vedením môjho

školiteľa.

V Bratislave, dňa 30.5.2021 ...................................................

Anna Rebeka Sojka

### Poďakovanie :

## Abstrakt

## Abstract

# Obsah

# Zoznam obrázkov

# Zoznam tabuliek

# Úvod

# Kapitola 1

# Východiská

## Vymedzenie základných pojmov

Cieľom tejto kapitoly je vybudovať pomocou logicky nadväzujúceho usporiadaného zoznamu definícií kľúčové pojmy, ktoré v práci rozšírene spomínam.

### Webová stránka

Online dostupné miesto na sieti, najmä na internete, sprístupňované prostredníctvom webového prehliadača a využívajúce hypertextový prenosový protokol alebo jeho zabezpečenú verziu, ktoré tvorí jednu vizuálnu obrazovku webového sídla, aj ak je zložené z viacerých rámov. [1]

### Webová aplikácia

Je taká aplikácia, ktorú nie je nutné inštalovať na zariadení užívateľa (počítač, tabliet, smartphone) a môžete ju spustiť z ktoréhokoľvek zariadenia pomocou webového prehliadača, pretože je spustená na strane servera. Vzhľadom k tomu, že je potrebné len prehliadač sa webová aplikácia niekedy nazýva tiež ako ľahký klient.

Výhody webových aplikácií:

* Nemusí sa inštalovať
* Užívateľ nemusí nič aktualizovať (aktualizácia prebieha na serveri)
* Potrebujete iba webový prehliadač
* Dáta sú uchovávané a zálohované na servera a sú prístupné odkiaľkoľvek
* Môžete k nim pristupovať odkiaľkoľvek z cudzieho počítača

Nevýhody webových aplikácií:

* Vyžadujú pripojenie na internet
* Niekedy pomalší tok dát a práce s aplikáciou (rýchlosť je závislá na kvalite pripojenia)
* Možno bezpečnostné riziká úniku dát v prípade nekvalitného poskytovateľa

[2]

### Databáza

Databáza je pojem pre štruktúrovaný súbor dát uložený na pamäťovom médiu. Databázy sú logicky, podľa určitých pravidiel, štruktúrované dáta a to vrátane systému ich vzájomných vzťahov a väzieb. Ide o báze dát, ktorú tvorí jedna alebo viac tabuliek s dátami. [3]

### PHP Frameworky

PHP frameworky (PHP Frameworks) sú frameworky napísané v programovacom jazyku PHP. Frameworky sú akousi nadstavbou jazyka PHP - obsahujú predpripravené celky, z ktorých sa aplikácia skladá rýchlejšie než za využitie čistého jazyka PHP. [4]

## Technológie

Táto kapitola menuje a popisuje technológie, ktoré som si zvolila pre vývoj aplikácie. Zároveň vysvetľuje tento výber, vzhľadom na rozsiahle možnosti, ktoré sú aktuálne dostupné s prihliadnutím na ich podporu a spoľahlivosť pre potenciálne širšiu verejnosť a s podmienkou, že ide o open-source licenciu.

Pre voľne dostupný Apache web server som používala XAMPP, populárnu voľbu pre Windows, pozostávajúci z MySQL, PHP a Perl. Textový editor v ktorom som pracovala bol Visual Studio Code.

### PHP

PHP je serverový skriptovací jazyk a výkonný nástroj na vytváranie dynamických a interaktívnych webových stránok.

PHP je široko používanou, bezplatnou a účinnou alternatívou pre konkurenciu, akou je napríklad Microsoft ASP.

PHP 7 je najnovšia stabilná verzia, [5] ktorá je zároveň aj medzi požiadavkami na server pre spoľahlivý chod mojej aplikácie, ako je uvedené v kapitole 1.3. Systémové požiadavky na server.

### Laravel

Laravel je open-source PHP framework, ktorý vytvoril Taylor Otwell v roku 2011. Aktuálne je zverejnená verzia číslo 8, pomocou ktorej som pracovala na tvorbe aplikácie. Ponúka najmä rozsiahlu podporu pre programovanie backendu, ale aj istú časť frontendu. Avšak pre vývoj dynamickej webovej aplikácie sa v praxi kombinuje väčšinou s JavaScript frameworkom Vue.js alebo prípadne s HTML a TypeScript frameworkom Angular. Pre frontend som zvolila Vue.js, keďže som sa dočítala z viacerých dokumentácií, že je pre tento účel najvhodnejší vďaka svojej kompatibilite s Laravelom.

Ale tento framework je silným nástrojom pre vývoj aj samostatne. Využíva:

* **MVC model a Eloquent ORM**

Architektonický vzor MVC model rozdeľuje kód do troch typov tried : Model, View a Controller. Vďaka tomu sa aj pre ďalšieho programátora kód javí ako čitateľnejší a prehľadnejší, keďže je potrebné dodržiavať konvencie.

Model (Laravelov objektovo-relačný mapovač Eloquent) reprezentuje dáta z tabuľky, View slúži na separovanie kódu pre používateľské rozhranie (vďaka čomu aj celý HTML kód je členený do viacerých komponentov) a Controller je možné označiť ako handler (či „manažér“) konkrétneho Modelu – napríklad GroupController (či „manažér skupiny“) bude : vytvárať, ukladať, editovať, mazať skupinu alebo skupiny z databázy a taktiež bude vracať triedy View, ktoré si používateľ vypýta ako profil skupiny, editačný formulár pre úpravy skupiny a podobne.

* **Artisan**

Laravel prichádza aj s vlastným rozhraním v príkazovom riadku. Pre zobrazenie všetkých funkcií, ktoré vie Artisan vykonať slúži príkaz

php artisan list

. Každý príkaz má aj pomocný popis, ktorý sa zobrazí po pridaní „help“ pred kľúčové slovo. Teda príkaz pre zobrazenie návodu pre

php artisan migrate

(príkaz pre vytvorenie databázy na základe kódu) by vyzeral nasledovne

php artisan help migrate

. Medzi populárne príkazy patrí aj

php artisan tinker

, ktorý spustí pre Artisan vlastné REPL (Read-Eval-Print-Loop) konzolové programovacie prostredie, ideálne pre izolované testovanie jednoduchších príkazov.

Pomocou takýchto Artisan príkazov sa väčšinou vytvárajú všetky triedy a nástroje projketu, keďže vďaka prepínačom sa navzájom prepájajú súvisiace triedy a upravujú vstupy pre štadardné funkcie. Taktiež sa vďaka tomuto spôsobu intuitívnejšie dodržujú konvencie, keďže sú triedy a s nimi aj funkcie predpripravené.

* **Query Builder**

Vytvára databázové dopyty pomocou vlastných metód, teda programátor nemusí písať príkazy priamo v SQL, pričom je chránený pred SQL injection útokmi (druhá strana využíva nedostatočne alebo úplne neošetrené príkazy a vkladá vlastný SQL kód).

* **Migrations**

Vďaka migráciám je možné vytvoriť všetky tabuľky databázy (vrátane integritných obmedzení a podobne) a pohodlne ich upravovať, zálohovať, presúvať a znovu využívať v ďalšom projekte.

Existuje aj mnoho ďalších funkcionalít, s čím sa spája hlavná výhoda Laravelu a tou je jeho podrobne vypracovaná mnohostranná dokumentácia a oficiálne video tutoriály, ktoré sú adresované rovnako začiatočníkom, ako aj skúseným programátorom.

### Vue.js

Vue je framework, ktorý sa sústreďuje na vytváranie používateľských rozhraní. Na rozdiel od iných frameworkov je Vue od základu navrhnutý tak, aby bol použiteľný samostatne, ale aj v kombinácií s inými knižnicami alebo existujúcimi projektmi. Taktiež je vhodný pre sofistikované jednostránkové aplikácie (Single-Page Applications), ak sa používa v kombinácii s modernými nástrojmi a podpornými knižnicami. [6] Pri tvorbe mojej aplikácie, som ho využila v spolupráci s Laravelom, kde poslúžil pri programovaní frontend časti kódu a zabezpečil potrebné dynamické vlastnosti. Vďaka Vue už nebolo potrebné zbytočné opätovné načítavanie stránky, teda zmeny v databáze sú okamžite zobrazené používateľovi. Dokonca v spolupráci s hosting službou Pusher umožnila kombinácia Laravel ako backend a Vue.js ako frontend zobrazovať zmeny na stránke v reálnom čase, ako je to napríklad nevyhnutné pre chat alebo vhodné sekciu pre komentáre.

Podobne ako framework Laravel disponuje Vue detailnou dokumentáciou a rastúcou popularitou.

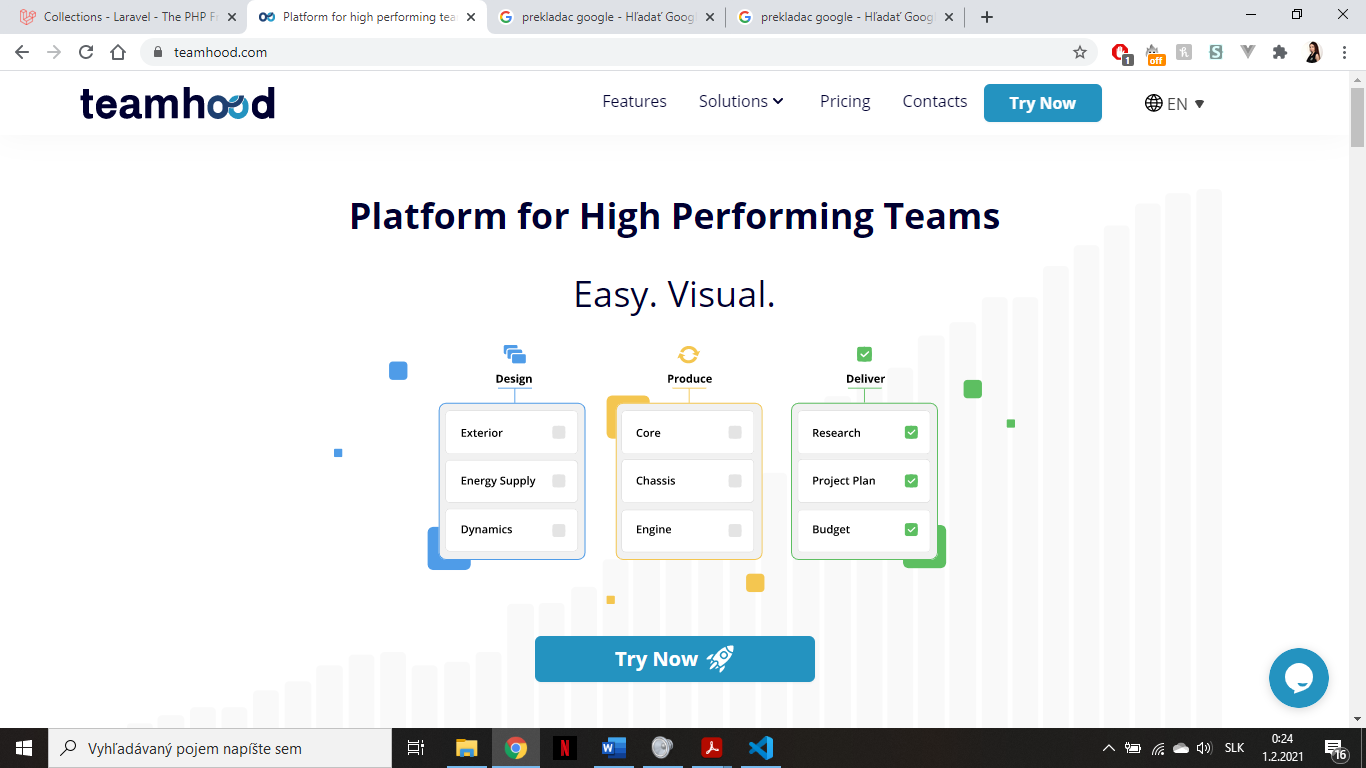
### Pusher Channels

Pusher Channels, teda Pusher kanály, poskytujú komunikáciu v reálnom čase medzi servermi, aplikáciami a zariadeniami. Kanály sa používajú pre tabuľky v reálnom čase, zoznamy používateľov v reálnom čase, mapy v reálnom čase, hry pre viacerých hráčov a pre mnoho ďalších typov komponentov vyžadujúcich neustálu aktualizáciu používateľského rozhrania.

Teda v prípade, že nastane v systéme nejaká zmena, Pusher umožní aktualizovať webové stránky, aplikácie a zariadenia. [7] Podobne aj napríklad v prípade udalosti na stránke. Keď používateľ pridá nový komentár, tak sa táto udalosť vysiela zvyšným používateľom v danej skupine a následne sa zobrazí bez opätovného načítavania stránky.

## Analýza podobných aplikácií

<https://blog.weekdone.com/best-team-management-apps/>



[www.teamhood.com](http://www.teamhood.com)

<https://startinfinity.com/>

<https://clockify.me/feature-list>

<https://trello.com/>

<https://www.meistertask.com/>

LITERATURA

[1] [95/2019 Z.z. - Zákon o informačných technológiách ... - SLOV-LEX (slov-lex.sk)](https://www.slov-lex.sk/pravne-predpisy/SK/ZZ/2019/95/20210101)

[2] Webová aplikácia (Web Application). In: ManagementMania.com [online]. Wilmington (DE) 2011-2021, 22.09.2015 [cit. 25.01.2021].

Dostupné z: <https://managementmania.com/sk/webova-aplikacia-web-application>

[3] (Database). In: ManagementMania.com [online]. Wilmington (DE) 2011-2021, 20.09.2016

[cit. 25.01.2021]. Dostupné z: <https://managementmania.com/sk/databaza>

[4] PHP frameworky. In: ManagementMania.com [online]. Wilmington (DE) 2011-2021, 13.01.2016 [cit. 26.01.2021].

Dostupné z: <https://managementmania.com/sk/php-frameworky>

[5] <https://www.w3schools.com/php/>

[6] <https://vuejs.org/>

[7] <https://pusher.com/docs>