SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Evidenčné číslo: FEI-16605-115306

VYTVORENIE WEBOVEJ STRÁNKY ZÁBAVNÉ

PROGRAMOVANIE PRE DETI

Bakalárska práca

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Evidenčné číslo: FEI-16605-115306

VYTVORENIE WEBOVEJ STRÁNKY ZÁBAVNÉ

PROGRAMOVANIE PRE DETI

Bakalárska práca

|  |  |
| --- | --- |
| Študijný program : | Aplikovaná informatika |
| Číslo študijného odboru: | 2511 |
| Názov študijného odboru: | 9.2.9 Aplikovaná informatika |
| Školiace pracovisko: | Ústav informatiky a matematiky |
| Vedúci záverečnej práce: | Mgr. Oľga Stašová, PhD |
| Konzultant ak bol určený: | Meno konzultanta |

Sem vložte zadanie z AIS

SÚHRN

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

|  |  |
| --- | --- |
| Študijný program : | Aplikovaná informatika |
| Vyberte typ práce | Vytvorenie webovej stránky |
| Autor: | Adam Dedinský |
| Vedúci záverečnej práce: | Mgr. Oľga Stašová |
| Konzultant ak bol určený: | Meno konzultanta |
| Miesto a rok predloženia práce: | Bratislava 2023 |

Cieľom práce bolo vytvorenie webovej stránky s názvom Zábavné programovanie pre deti. Bakalárska práca obsahuje úvod do problematiky so súvisiacimi problémami pri vytváraní webových stránok, analýzu stránok s podobnou témou, prehľad technológii, ktoré je možné využiť, vyhodnotenie prínosu webovej stránky pre užívateľov, pre ktorých je určená. Na webovej stránke sa nachádza súbor s viacerými úlohami, ktoré užívateľ môže vypracovať. Úlohy sú rozdelené podľa obťažnosti. K vypracovaniu každej úlohe sa na stránke nachádza dokumentácia s potrebným vysvetlením syntaxe pre vypracovanie úlohy. Implementoval som možnosť registrácie aby každý užívateľ vedel aké ma hodnotenie a vedel sa porovnávať s ďalšími začínajúcimi programátormi. K tomu je vytvorená podstránka Rebríček, kde budú zobrazené užívateľské mená najlepších desiatich programátorov, ktorí majú najviac získaných bodov. (Doplniť či sa mi podarilo splniť,)

Kľúčové slová: PHP, HTML, CSS, JavaScript, C++

ABSTRACT

SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA

FACULTY OF ELECTRICAL ENGINEERING AND INFORMATION TECHNOLOGY

|  |  |
| --- | --- |
| Study Programme: | Applied Informatics |
| Bachelor Thesis: | Create a website Fun programming  for kids |
| Autor: | Adam Dedinský |
| Supervisor: | Mgr. Oľga Stašová, PhD |
| Consultant: | Meno konzultanta |
| Place and year of submission: | Bratislava 2023 |

The goal of the work was to create a website called "Fun Programming for Kids." The bachelor's thesis includes an introduction to the issues related to web development, analysis of websites with a similar theme, an overview of technologies that can be used, evaluation of the website's benefits for the users it is intended for. The website contains a set of tasks that users can complete, with tasks categorized by difficulty. For each task, there is documentation on the website with necessary explanations of the syntax for task completion. I implemented a registration feature so that every user could see their own ratings and compare themselves with other beginner programmers. For this purpose, a "Rebríček" subpage was created, where the usernames of the top ten programmers with the most points will be displayed. (To be added)

Key words: PHP, HTML, CSS, JavaScript, C++

Vyhlásenie autora

Podpísaný Adam Dedinský čestne vyhlasujem, že som Bakalársku prácu Vytvorenie webovej stránky Zábavné programovanie pre deti vypracoval na základe poznatkov získaných počas štúdia a informácií z dostupnej literatúry uvedenej v práci.

Uvedenú prácu som vypracoval pod vedením Mgr. Oľga Stašová, PhD.

V Bratislave dňa 19.10.2023

..................................................

podpis autora

Poďakovanie

Sem môžete vložiť ďakovný text. Spravidla sa ďakuje vedúcemu práce, prípadne konzultantovi. Poďakovanie nie je povinná súčasť práce.

Obsah

[Úvod 1](#_Toc148625901)

[1 Budúcnosť programovania 2](#_Toc148625902)

[2 Front end 3](#_Toc148625903)

[2.1 HTML 3](#_Toc148625904)

[2.2 CSS 5](#_Toc148625905)

[2.3 JavaScript 5](#_Toc148625906)

[2.4 Framework 6](#_Toc148625907)

[3 BACK END 7](#_Toc148625908)

[3.1 JavaScript 7](#_Toc148625909)

[3.2 PHP 7](#_Toc148625910)

[3.2.1 Laravel 7](#_Toc148625911)

[3.3 MySQL 8](#_Toc148625912)

[3.4 Phpmyadmin 8](#_Toc148625913)

[3.5 XAMPP 8](#_Toc148625914)

[1.4.1 Postup vkladania citácie 11](#_Toc148625915)

[Záver 14](#_Toc148625916)

[Zoznam použitej literatúry 15](#_Toc148625917)

[Prílohy I](#_Toc148625918)

Je potrebné aktualizovať pole obsahu, aby sa zobrazili aktuálne čísla strán.

Zoznam obrázkov a tabuliek

[Obrázok 1 - Značky HTML 4](file:////Users/mac/Desktop/Bakalarka/dedinsky_adam_bc_praca.docx#_Toc148630377)

[Obrázok 2 – Webová stránka s CSS 5](file:////Users/mac/Desktop/Bakalarka/dedinsky_adam_bc_praca.docx#_Toc148630378)

[Obrázok 3 – Webová stránka bez CSS 6](file:////Users/mac/Desktop/Bakalarka/dedinsky_adam_bc_praca.docx#_Toc148630379)

[Obrázok 4 - Vytvorenie tabuľky v databáze 8](file:////Users/mac/Desktop/Bakalarka/dedinsky_adam_bc_praca.docx#_Toc148630380)

Tabuľka 1 Popis tabuľky.............................................................................strana

Zoznam skratiek a značiek

HTML – Hypertext Markup Language

CSS – Cascading Style Sheets

Úvod

Témou bakalárskej práce je webstránka programovanie pre deti. Túto tému som si vybral na základe niekoľkých dôvodov, ktoré uvediem neskôr. Ide o webstránku, kde používatelia získajú základné znalosti ohľadom programovania v jazyku C++, avšak s tým sa spájajú všeobecné znalosti programovania.

Hlavný dôvod pre výber tejto témy je, že na Slovensku máme len veľmi málo toho, čo by viedlo deti k myšlienke, aby sa raz chceli stať programátormi. Určite nechcem znevažovať predmet Informatika na základnej škole, no z vlastných skúsenosti viem, že tam som neprepadol do sveta technológii.

V dnešnej dobe žijeme v dobe internetu, čo nie je online chýba čiarka ako keby ani neexistovalo. Ja to nepovažujem za problém ale za výhodu. Rýchlo a jednoducho sa dostaneme k informáciám, ktoré hľadáme. Ale vždy je čo zlepšovať v technologickom svete. A práve preto je potrebné k tomu viesť mladšiu generáciu, nech Slováci nie sú pozadu oproti zvyšku sveta, a taktiež sa stanú súčasťou celosvetových technologických korporácii.

Tak som sa dostal k cieľu mojej práce, oboznámiť so svetom technológii práve mladšiu generáciu aj iným spôsobom ako len tým používateľským. Keď budú vypracovávať rôzne zadania, ktoré sa budú nachádzať na webovej stránke, tak sa stanú malými programátormi a možno ich to začne baviť, to ich môže priviesť aj k iným a zložitejším kurzom, tak sa z nich postupne môžu stať plnohodnotní programátori už v relatívne nízkom veku.  
 Prečo práve jazyk C++? Je to môj obľúbený programovací jazyk, a aj keď osobne viem, že nie je jednoduchý, no príde mi prehľadný, keď sa dodržujú dané pravidlá písania kódu. A to považujem za podstatnú vec, aby sa neskúsený začiatočník nezľakol, keby sa nevyznal v kóde, ktorý píše. Taktiež sa dajú v ňom písať jednoduché kódy, ktoré budú postupne nadobúdať vyššiu obťiažnosť.

# Budúcnosť programovania

Programovanie je kľúčovou zručnosťou pre budúcnosť. Stále viac a viac sa technológie dostávajú do našich každodenných aktivít. Táto kľúčová zručnosť mení už desaťročia spôsoby, ako sa vzdelávame, zabávame, komunikujeme a podobne. Každým rokom prichádzajú novinky a inovácie, ktoré nám uľahčujú život.

S týmto rastúcim významom digitálneho sveta je programovanie v dnešnom svete nevyhnutné v každom odvetví. Aj napriek tomu tu máme stále mnoho vecí a aktivít, ktoré nie sú zatiaľ pripravené k digitalizácii. Už niekoľko storočí je známe, ako si ľudia chcú prirodzene uľahčiť prácu. A práve aj to sa týka budúcnosti programovania. Čo raz viacej vecí sa budeme snažiť si uľahčiť pomocou programovania.

Aktuálne je budúcnosť programovania najviac spojená s umelou inteligenciou. Tá prišla na svet len nedávno a vytvorila prevrat vo svete programovania [1]. Stále je len vo vývine a možno aj navždy bude, a preto potrebujeme vo svete nových programátorov. Programovanie je okrem umelej inteligencie spájané aj so starostlivosťou o ľudské zdravie. Každý štát s vyspelým zdravotníctvom využíva najnovšie technológie na liečenie ľudí. Aj napriek tomu tu máme zákerné choroby, na ktoré umiera nespočetne veľa ľudí. Na tieto choroby sa snažia vedci vynájsť lieky, no na to vynájdenie potrebujú samozrejme už spomínané technológie. Takže ich vývoj sa určite nikdy nezastaví a budú tu pri nás pre zlepšenie kvality života.

Celkovo vzaté je to teda neodlúčiteľná súčasť budúcnosti, ktorá ovplyvňuje spoločnosť, ekonomiku, zdravotníctvo, vzdelávanie... Preto je dôležité viesť ľudí k tejto zručnosti, aby mohli zlepšovať všetky tieto oblasti.

# Front end

Front end je označovaný ako používateľská časť webovej stránky. Keď používateľ navštívi webovú stránku, tak prvý kontakt má práve s front endom. Všeobecne zahŕňa táto časť všetko od rozloženia, dizajnu, a používateľského rozhrania až po interaktívnu časť webovej stránky. Je to jedna z najdôležitejších častí webových stránok, s ohľadom na konkurenciu je cieľom front endu zaujať a udržať si používateľov. Cieľom každého programátora tejto časti by malo byť vytvorenie vizuálne príťažlivého a funkčného prostredia, taktiež by mali byť všetky funkčné časti intuitívne, jednoduché na pochopenie. Výnimočný a funkčný front end nielenže zlepšuje používateľskú skúsenosť, ale napomáha k budovaniu úspešnej webovej stránky. Je to prvý krok k tomu, aby sa používateľ vrátil a  neuprednostnil konkurenčnú webovú stránku.

V dnešnej dobe je to ale trocha zložitejšie, ako sa zdá. Na trhu existuje veľký počet elektronických zariadení, cez ktoré vieme zobrazovať webstránky. Tu nastáva problém pre vývojárov, musia prispôsobovať webové stránky na rôzne veľkosti displejov. Túto vlastnosť nazývame responzívnosť. Ak je teda stránka responzívna, tak sa prispôsobuje podľa šírky a výšky displeja, na ktorom sa zobrazuje, tým vie vývojár docieliť správne zobrazenie webovej stránky a taktiež uchovať funkčnosť všetkých elementov, ktoré sa na nej nachádzajú.

Dnes už máme dostupných mnoho technológii pre vývoj používateľskej časti webovej stránky. Cieľom tejto časti bude oboznámiť sa s niektorými možnosťami, ktoré som aj ja využil pri vytváraní webovej stránky Programovanie pre deti.

## HTML

HTML je značkovací jazyk [2]. Využíva sa na vytváranie každej webovej stránky, aj keď je stránka vytváraná pomocou alternatív ako WordPress, Shopify... tak vkladaním elementov v týchto nástrojoch sa vytvára HTML kód. Je to jednoduchšie a  nie je potrebné ovládať značkovací jazyk HTML, pokiaľ teda nie sú potrebné zložitejšie úpravy, tam už je nutné mať znalosť HTML. HTML je štruktúrovaný do usporiadaných elementov, ktoré sú označované pomocou značiek tzv. tagov. Elementy a ich obsah je teda obklopený tagmi z ľavej a pravej strany. Okrem toho existujú aj nepárové tagy, kde nie je potrebný A screen shot of a computer screen

Description automatically generatedukončovací tag.

Obrázok 1 - Značky HTML

Na Obrázku 1 vidíme základné a najpoužívanejšie tagy. Okrem teda základných ako sú napríklad <h1></h1>, <p></p> sa tam nachádzajú aj takzvané vnorené elementy. Tie umožňujú vkladať tagy do iných tagov, to slúži na vytváranie komplexných stránok. Pre príklad to je <ol></ol> (usporiadaný zoznam), v ktorom sa nachádzajú tagy <li> (položka v usporiadanom zozname).

HTML je jazyk, ktorý sa neustále vyvíja. Nové verzie prichádzajú s vylepšeniami a novými funkciami. Aktuálna verzia je HTML5, ktorá priniesla podporu pre multimédia, rozšírenie možnosti pre dynamický obsah a iné.

## CSS

CSS po slovensky nazývame kaskádové štýly. Je používané pre definovanie vizuálneho vzhľadu webových stránok. Obsahuje kód pre štýl, farbu a veľkosť písma, odsadenie elementov a mnoho ďalších vecí. Umožňuje vývojárom vytvoriť unikátny a profesionálny dizajn stránky, ktorý sa aplikuje na webstránku.

História CSS sa začala písať už v deväťdesiatych rokoch dvadsiateho storočia, kedy začal internet získavať popularitu. Vtedy existovali statické stránky, ktoré by dnes na nás pôsobili neprofesionálne a jednoducho. Všetko bolo definované iba v kóde HTML, to znamenalo opakujúci sa a neprehľadný kód. Prvýkrát bolo CSS predstavené v roku 1996 [3], vtedy umožňovalo oddeliť štýlové pravidlá od samotného obsahu HTML. To malo za následok lepšie a jednoduchšie spracovanie webových stránok. Postupne prichádzali vylepšenia, ktoré pridávali nové funkcie ako napríklad podpora pre animácie, spomínaný responzívny dizajn a všetko ostatné. čo nám je už dnes známe.

A screenshot of a computer

Description automatically generatedDôležitá časť je kaskádovanie, to znamená, že je možné použiť štýly všeobecne pre celú webstránku, a potom v prípade potreby je možné ich prepisovať pre konkrétne časti webovej stránky.

Obrázok 2 – Webová stránka s CSS

A screenshot of a computer

Description automatically generatedCSS je teda prepojené s HTML, a v CSS pridávame atribúty, ktoré menia vzhľad a správanie tagov z HTML. Pre daný dizajn tagov vytvárame v CSS triedy, ktoré obsahujú spomínané atribúty. Tak docielime napríklad iný dizajn pre nadpis h1 a h2. Taktiež vieme pomocou CSS určiť, či prvky majú byť usporiadané horizontálne alebo vertikálne.

Obrázok 3 – Webová stránka bez CSS

## JavaScript

JavaScript je navrhnutý pre internetové aplikácie. Je súčasťou ako front endu tak aj back endu. V časti front endu slúži pre dynamickosť webovej stránky, ovládanie interaktívnych prvkov HTML a podobne. V podstate mení statické prvky na dynamické, ktoré potom vedia reagovať na rôzne podnety ako je klik, vloženie textu... Vieme v ňom implementovať cykly, podmienky, funkcie, polia a podobne.

## Framework

Framework je predpripravený súbor s nástrojmi pre jednoduchšie vytváranie webových stránok a webových aplikácií. Poskytuje už hotové riešenia pre bežné úlohy a problémy. Z toho vyplýva jeho hlavné využitie a to je ušetriť programátorom čas a úsilie. Frameworky môžu byť určené pre viaceré účely, napríklad pre webový vývoj tu máme Bootstrap [4], pre mobilný vývoj React Native [5] a mnohé ďalšie. Ďalšou výhodou týchto súborov s nástrojmi je, že programátor sa vyhne zbytočným chybám, ktoré by mohli vzniknúť bez ich využitia.

# BACK END

Vo všeobecnosti sa webstránka skladá z dvoch častí – front end a back end. Front end je tiež známy ako klientska časť, ktorú vidíme v prehliadači. Back end, serverová časť, je všetko čo sa deje v pozadí, a táto časť nie je viditeľná [6]. Back end vo všeobecnosti zahŕňa komunikáciu medzi klientom a serverom, komunikáciu s databázou a spracováva používateľské interakcie aby mohol na ne reagovať.

V súčasnej dobe sa používajú rôzne back end technológie. Výber správnej technológie závisí od požiadaviek daného softvéru. Najviac používané programovacie jazyky pre túto časť sú JavaScript, PHP, Python, C#.

## JavaScript

Pre mnohých vývojárov slúži JavaScript pre vývoj front end časti. Využívajú ho na vytvorenie dynamickej časti web stránok, pretože je jednou z kľúčových technológii, ktoré poháňajú moderný web [7]. Napriek tomu sa dnes využíva aj v časti back end, prostredníctvom platformy Node.js, ktorá je JavaScript-ové prostredie. Node.js umožňuje vývojárom pracovať a manipulovať s dátami z front endu. Hlavné funkcie, ktoré poskytuje pre back end:

* Spracovanie HTTP požiadaviek a odpovedí,
* Prístup k súborovému systému servera,
* Komunikácia s databázou.

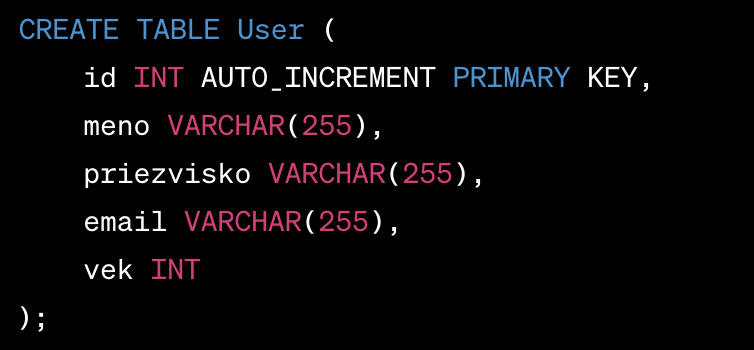
## PHP

PHP je skriptovací jazyk, ktorý generuje HTML obsah. Tento vygenerovaný obsah odosiela na webový prehliadač. Je jednoduchý na integrovanie a jeho hlavný účel je vytvárať dynamický obsah tak, že generuje HTML na základe dát, ktoré sú v danom okamihu k dispozícii. To umožňuje vývojárom webových stránok reagovať na používateľské interakcie a zobrazovať rôzne informácie.

### Laravel

Laravel je bezplatný a open-source PHP framework, ktorý poskytuje súbor nástrojov na vytváranie moderných PHP aplikácií [8]. Laravel si získal pozornosť vďaka jednoduchej syntaxi a zabudovaným balíčkom funkcií a nástrojov, ktoré uľahčujú rôzne úlohy. Tým pádom netreba kód písať úplne od základov.

## MySQL

Databázy predstavujú základné úložisko údajov pre všetky softvérové aplikácie [9]. MySQL samotný nie je programovacím jazykom. Je to typ softvéru, ktorý umožňuje a uľahčuje tvorbu riešení pre rôzne problémy. V MySQL môžeme vytvárať, upravovať a mazať databázy. V týchto databázach sa nachádzajú tabuľky, ktoré definujú štruktúru dát pomocou stĺpcov a riadkov. Vývojári definujú názvy stĺpcov, ich dátové typy a ďalšie vlastnosti, ktoré určujú, aké dáta sa budú ukladať do tabuliek. Každá tabuľka obsahuje primárny kľúč, ktorý slúži na jednoznačnú identifikáciu záznamov v danej tabuľke. Pre prepojenie tabuliek v databáze sa používa cudzí kľúč.

Obrázok 4 - Vytvorenie tabuľky v databáze

Vzťahy medzi tabuľkami sú definované rôznymi násobnosťami. Existujú štyri základné typy vzťahov:

1. One-to-One *(jeden k jednému)*: Tento vzťah mapuje jeden záznam z primárnej tabuľky na jeden záznam zo sekundárnej tabuľky.
2. Many-to-Many *(viacerí k viacerým)*: Vzťah many-to-many spojuje viaceré entity z dvoch rôznych tabuliek.
3. One-to-Many *(jeden k viacerým)*: Tento vzťah prepojí jeden záznam z primárnej tabuľky s viacerými záznamami zo sekundárnej tabuľky.
4. Many-to-One *(viacerí k jednému)*: Posledný vzťah, many-to-one, spojuje viaceré záznamy z primárnej tabuľky s jedným záznamom zo sekundárnej tabuľky.

## PhpMyadmin

PhpMyAdmin je bezplatný softvérový nástroj napísaný v PHP, obsahuje administráciu MySQL a MariaDB [10]. Jeho hlavnou funkciou je umožniť správcovi jednoduchý a prehľadný prístup k práci s databázou. Používatelia webových stránok nemajú prístup k databázam, ktoré daná webová stránka využíva. Slúži iba pre vývojárov. PhpMyAdmin má mnoho funkcií, medzi tie hlavné patria:

* Vytváranie a správa databáz: S phpMyAdmin môže vývojár vytvárať, spravovať, vymazávať databázy.
* Správa tabuliek: Tak ako je možné spravovať databázy, tak to isté je možné s tabuľkami. V prípade potreby pridania/vymazania ďalšieho stĺpca alebo zmenenie typu dát, nie je nutné vytvárať novú tabuľku a nahradzovať starú, stačí upraviť nastavenia pôvodnej.
* Import a export: PhpMyAdmin umožňuje správcovi exportovať a importovať údaje z databázy, okrem toho môže importovať a exportovať celú databázu.
* Dokumentácia a komunita: PhpMyAdmin je jednoduchý a veľmi dobre zdokumentovaný. Má aktívnu a veľkú komunitu, čo znamená, že je na internete mnoho návodov pre riešenie problémov.

## XAMPP

XAMPP je multiplatformový webový server, ktorý je bezplatný a open-source [10]. Základná vec, ktorú XAMPP robí, je že vytvára lokálny server. To znamená, že server je len na počítači, kde sa spustil server cez softvér XAMPP. Slúži vývojárom aby mohli jednoducho a rýchlo testovať webové stránky, na ktorých pracujú bez nutnosti ich zverejňovať na iné servery. S tým súvisí izolácia od skutočného internetu. Keď sa testuje webová stránka pomocou XAMPP, tak nie je verejná a  žiadny používateľ sa k nej nedostane. Vďaka tomu je webová stránka dobre zabezpečená, keďže reálne sa nenachádza nikde inde ako na tom danom počítači. Tým, že to je na počítači, kde vývojár pracuje, dostáva rýchlu spätnú väzbu. To mu dovoľuje rýchlo opravovať chyby alebo vyvíjať nové funkcie.

* 1. Analýza problému – Súčasný stav riešenej problematiky

V časti Analýza problému autor uvádza súčasný stav riešenej problematiky doma i v zahraničí, dostupné informácie a poznatky týkajúce sa danej témy. Zdrojom pre spracovanie sú aktuálne publikované práce domácich a zahraničných autorov. Základné definície a formalizmy potrebné na riešenie problematiky.

* 1. Opis riešenia

Časť Opis riešenia jasne, výstižne a presne charakterizuje predmet riešenia. Súčasťou sú aj rozpracované čiastkové ciele, ktoré podmieňujú dosiahnutie hlavného cieľa. Ak je práca implementačná, tak jej súčasťou musí byť aj softvérová špecifikácia požiadaviek, návrh, implementácia, overenie riešenia. Treba podľa možností vychádzať zo známych prístupov. Táto časť práce závisí od konkrétneho zadania. Je dôležité prezentovať návrhové rozhodnutia, alternatívy, ktoré sa zvažovali pri riešení a samotný návrh riešenia zadaného problému. Štruktúra textu by mala vychádzať zo zadanej úlohy, ktorá sa rieši. Najmä v tejto časti študent preukazuje svoj originálny prístup k riešeniu problémov a kritické myslenie.

Súčasťou môže byť metodika práce a metódy skúmania, ktoré spravidla obsahujú:

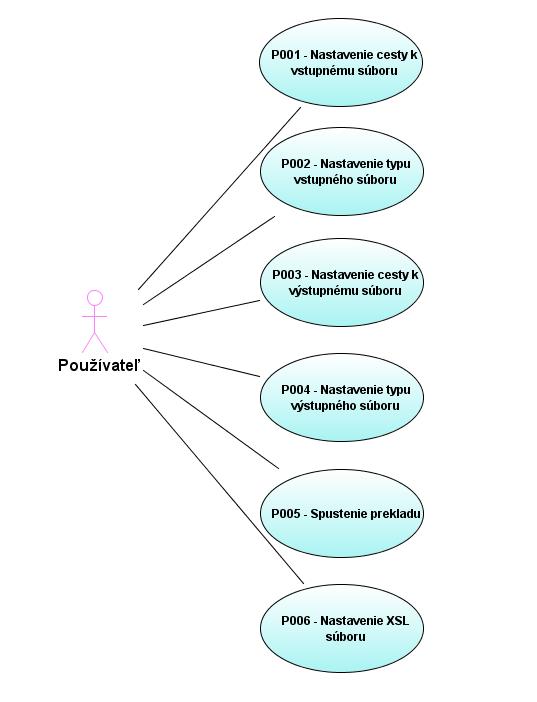
a) charakteristiku objektu skúmania

b) pracovné postupy

c) spôsob získavania údajov a ich zdroje

d) použité metódy ich vyhodnotenia a interpretácie výsledkov

Implementácia musí byť otestovaná. Výsledok musí byť porovnaný s inými riešeniami.



* 1. Zhodnotenie

Výsledky (vlastné postoje alebo vlastné riešenie vecných problémov), ku ktorým autor dospel, sa musia logicky usporiadať a pri popisovaní sa musia dostatočne zhodnotiť. Zároveň sa komentujú všetky skutočnosti a poznatky v konfrontácii s výsledkami iných autorov. Ak je to vhodné, výsledky práce a diskusia môžu tvoriť samostatné časti ZP.

* 1. Citácie

Citácia alebo citovanie je skrátené označenie citovaného zdroja (dokumentu) v texte práce podľa niektorej metódy odporučenej normou STN ISO 690. Citácia slúži na spojenie citovaného miesta so záznamom o citovanom dokumente. Tento záznam je potom položkou v zozname bibliografických odkazov. Citácia slúži aj na spojenie citovaného miesta so skráteným záznamom o citovanom dokumente umiestneným napr. pod čiarou na príslušnej strane textu. VZOR CITÁCIE (1)

* + 1. Postup vkladania citácie

Na karte REFERENCES si zvolíme štýl citácie ISO-690- Numerical references. Následne klikneme na Manage sources a kliknutím na new otvoríme okno na vloženie nového bibliografického odkazu. Z rolovacieho menu vľavo hore vyberieme typ dokumentu, ktorý budeme citovať a vyplníme všetky známe údaje o zdroji citácie. Potvrdíme a vrátime sa späť na písanie dokumentu. Na karte references klikneme na Insert Citation a vyberieme citovaný zdroj. V dokumente sa objaví číselný odkaz. VZOR CITÁCIE (1)

* 1. Špeciálne požiadavky

Ak je súčasťou vašej práce vytvorenie softvéru je nutné k tomuto softvéru vytvoriť dokumentáciu (technickú dokumentáciu, užívateľskú príručku) a pripojiť ju ku práci vo forme prílohy. Ak je dokumentácia rozsiahla, je vhodnejšie ju pridať ako prílohu na CD/DVD. Ak je kratšia, tak je vhodné ju pridať aj v tlačenej forme.

1. Popis šablóny

V šablóne sú použité viaceré druhy polí. Pevné polia nedovoľujú prepisovať ich obsah. Naopak polia, ktorých text je vyznačený červeným písmom musí byť zmenený, alebo vymazaný. V šablóne sa nachádzajú selektívne polia, ktoré umožňujú výber z viacerých variant. S poľami sa dá pracovať na karte vývojár, ktorú je možné vložiť v nastaveniach.

* 1. Popis nastavenia strany

OKRAJE: hore 3cm, dole 3cm, vľavo 3cm, vpravo 2,5cm, orientácia: na výšku

PAPIER: typ: A4, šírka: 21cm, výška:29,7cm,

ROZLOŽENIE: hlavička: 1,5cm, päta: 1,5cm, zvislé zarovnanie: hore

* 1. Popis nastavenia štýlov

NADPIS 1.ÚROVNE: založiť na: žiadnom, štýl nasledujúceho odseku: základný, Písmo: Times New Roman, 22 b, Tučné, Vľavo, Riadkovanie: jednoduché, Medzera Za: 16b, Kontrola osamotených riadkov, Zlom strany pred odsekom, Zviazať s nasledujúcim, Zviazať riadky dohromady, Viacúrovňové + Úroveň: 1 + Štýl číslovania: 1, 2, 3, … + Číslovať od: 1 + Zarovnanie: Vľavo + Zarovnať na: 0 cm + Zarážka: 1,27 cm, Štýl: Prepojené, Automaticky aktualizovať, Zobraziť v galérii štýlov

NADPIS 2.ÚROVNE: založiť na: žiadnom, štýl nasledujúceho odseku: základný, Písmo: Times New Roman, 16 b, Tučné, Zarážka: Vľavo: 0 cm, Opakovaná zarážka: 0,6 cm, Vľavo, Riadkovanie: jednoduché, Medzera Pred: 16 b, Za: 14 b, Kontrola osamotených riadkov, Zviazať s nasledujúcim, Zviazať riadky dohromady, Viacúrovňové + Úroveň: 2 + Štýl číslovania: 1, 2, 3, … + Číslovať od: 1 + Zarovnanie: Vľavo + Zarovnať na: 0 cm + Zarážka: 0,6 cm, Štýl: Prepojené, Automaticky aktualizovať, Zobraziť v galérii štýlov

NADPIS 3.ÚROVNE: založiť na: žiadnom, štýl nasledujúceho odseku: základný, Písmo: Times New Roman, 14 b, Tučné, Zarážka: Vľavo: 0 cm Opakovaná zarážka: 0,6 cm, Vľavo, Riadkovanie: jednoduché, Medzera Pred: 14 b Za:14 b, Kontrola osamotených riadkov, Zviazať s nasledujúcim, Zviazať riadky dohromady, Viacúrovňové + Úroveň: 3 + Štýl číslovania: 1, 2, 3, … + Číslovať od: 1 + Zarovnanie: Vľavo + Zarovnať na: 0 cm + Zarážka: 0,6 cm, Štýl: Prepojené, Zobraziť v galérii štýlov

ZÁKLADNÝ: založiť na: žiadnom, štýl nasledujúceho odseku: základný, Písmo: Times New Roman, 12 b, Zarážka: Prvý riadok: 0,8 cm, Podľa okraja, Riadkovanie: 1,5 riadka, Kontrola osamotených riadkov, Štýl: Prepojené, Automaticky aktualizovať, Zobraziť v galérii štýlov

POPIS: Písmo: 9 b, Kurzíva, Farba písma: Text, Riadkovanie: jednoduché, Medzera

Za: 10 b, Štýl: Skryť, kým nie je použité, Zobraziť v galérii štýlov, Priorita: 36, Podľa: Normálny

Použitie: na popis obrázkov, tabuliek a grafov

NADPIS NEČÍSLOVANÝ: Bez odrážok a číslovania, Štýl: Prepojené, Zobraziť v galérii štýlov Podľa: Nadpis 1.úrovne

**Záver**

V závere je potrebné v stručnosti zhrnúť dosiahnuté výsledky vo vzťahu k stanoveným cieľom.

**Zoznam použitej literatúry**

**Prílohy**

Príloha A: Štruktúra elektronického nosiča . . . . . . . . . . . . . . . . . . . . . . . . . II

Prílohy sú „číslované“ písmenami A, B, C...

Príloha A: Štruktúra elektronického nosiča

Štruktúra elektronického nosiča (CD, DVD, atď.) s kompletnou digitálnou verziou tlačenej formy práce, vrátane príloh, funkčných zdrojových kódov, programov (aplikácií) pripravených na inštalovanie a iných, vo všeobecnosti ťažko opísateľných ale potrebných častí. Elektronický nosič musí mať obal, pomocou ktorého sa pevne pripevní do práce. Nosič musí mať popis obsahu a meno autora.