**ř Vyšší odborná škola**



**Vyšší odborná škola**

**a Střední průmyslová škola elektrotechnická**

**Plzeň, Koterovská 85**

**Vyšší odborná škola**

**Semestrální práce**

**Prodej zboží**

**Plzeň květen 2024 Tomáš Koller**

Anotace

Cielom semestrálnej práce bolo vytvoriť internetový obchod pomocou programovacích jazykov HTML5, CSS3, JavaScript, PHP a MySQL. Autor sa snažil v práci bez používania frameworkov vytvoriť internetový obchod pre predaj jedla a suplementov ktoré sa nachádzajú v databázi na serveri a bezpečne uložiť uživatelské údaje do nej. Práca je členená do kapitol, podkapitol a praktickej časti. Výsledkom práce je preukázať vedomosti vo fullstack vývoji webových aplikácií.

Obsah

[Anotace 2](#_Toc168346768)

[Úvod 5](#_Toc168346769)

[Štruktúra súborov 6](#_Toc168346770)

[Drátené modely stránok 7](#_Toc168346771)

[Drátený model page\_landing.php 7](#_Toc168346772)

[Drátený model page\_login.php 8](#_Toc168346773)

[Drátený model page\_contactus.php 9](#_Toc168346774)

[Drátený model page\_food.php 10](#_Toc168346775)

[Drátený model page\_suplements.php 11](#_Toc168346776)

[Drátený model page\_register.php 12](#_Toc168346777)

[Drátený model page\_admin.php 13](#_Toc168346778)

[Funkcie 14](#_Toc168346779)

[SendMessage($conn) 14](#_Toc168346780)

[Parametre: 14](#_Toc168346781)

[Popis: 14](#_Toc168346782)

[loginUser($conn) 14](#_Toc168346783)

[Parametre: 15](#_Toc168346784)

[Popis: 15](#_Toc168346785)

[logoutUser() 16](#_Toc168346786)

[Parametre: 16](#_Toc168346787)

[Popis: 16](#_Toc168346788)

[registerUser($conn) 16](#_Toc168346789)

[Parametre: 16](#_Toc168346790)

[Popis: 16](#_Toc168346791)

[getSuplements($conn) 17](#_Toc168346792)

[Parametre: 17](#_Toc168346793)

[Popis: 17](#_Toc168346794)

[getFoods($conn) 18](#_Toc168346795)

[Parametre: 18](#_Toc168346796)

[Popis: 18](#_Toc168346797)

[getAllOrders($conn) 19](#_Toc168346798)

[Parametre: 19](#_Toc168346799)

[Popis: 19](#_Toc168346800)

[Komponenty 20](#_Toc168346801)

[Component\_hlava.php 20](#_Toc168346802)

[Component\_nav.php 20](#_Toc168346803)

[Databáza 20](#_Toc168346804)

[Relačný model MySQL databáze 20](#_Toc168346805)

[Databáza zakaznik 20](#_Toc168346806)

[Databáza Objednavky 20](#_Toc168346807)

[Databáza zbozi 20](#_Toc168346808)

[Databáza druhyzbozi 20](#_Toc168346809)

[Databáza kontakt 20](#_Toc168346810)

[Vzťahy medzi tabuľkami 20](#_Toc168346811)

[Použitie databázového modelu 20](#_Toc168346812)

[JavaScript 20](#_Toc168346813)

[Main\_script.js 20](#_Toc168346814)

[navOpen() 20](#_Toc168346815)

[navClose() 20](#_Toc168346816)

[kosikOpen() 21](#_Toc168346817)

[kosikClose() 21](#_Toc168346818)

[CSS 22](#_Toc168346819)

[Styles.css 22](#_Toc168346820)

[Fotky 22](#_Toc168346821)

Úvod

Webové aplikácie umožňujú v dnešnej dobe nám z pohodlia domova vyriešiť komplexné životné potreby kvôli ktorým by sme museli opúšťať domov. Vďaka moderným technológiam a moderným programovacím jazykom sme schopný funkčne zautomatizovať celú problematiku nakupovania produktov a menežovanie dostupných položiek a platby. Na to aby daná webová aplikácia bola použitelná by mala byť prehľadná pre uživateľa a nemala by ho navigácia po nej frustrovat. Malo by byť jasné s ktorými prvkami sa na stránke dá interaktovať a ktoré slúžia iba ako vizuál. Kontrast medzi pozadím a textom pre uživateľa by mal byť dostatočný na to aby nebolo pre neho tažké text čítať či ho nájsť na pozadí.

Na prvotný dizajn webovej stránky som použil program pre dizajn webových aplikácií Figma. Po zrealizovanie predstavi o dizajne a výbere farieb pre stránku som zrealizoval celý projekt vo Visual Studio Code ktorý za pomoci rozšírení uľahčil celú prácu s kódom a jeho formatovaním.

Pre vytvorenie štruktúry webovej aplikácie bol použitý značkovací jazyk HTML5 ktorý webovej aplikácii dodá štruktúru prvkov ktoré následne som nadizajnoval a umiestnil na ich pozície pomocou CSS3. JavaScript primárne slúži v mojej práci pre aktivovanie a deaktivovanie interaktívnych prvkov v reálnom čase. PHP som použil na premostenie front-endu s databázov a teda na komunikáciu medzi nimi. Ako jazykový model pre databázu som použil MySQL ktorý umožní štrukturované uloženie a uschovávanie dát na serveri.

Štruktúra súborov

Súbory som si menom rozdelil na komponenty, funkcie, uživateľské stránky, stránky webu a doplnkové pre ľahšiu orientáciu medzi nimi a ich účelom.

Obrázok, na ktorom je text, snímka obrazovky, softvér, multimediálny softvér

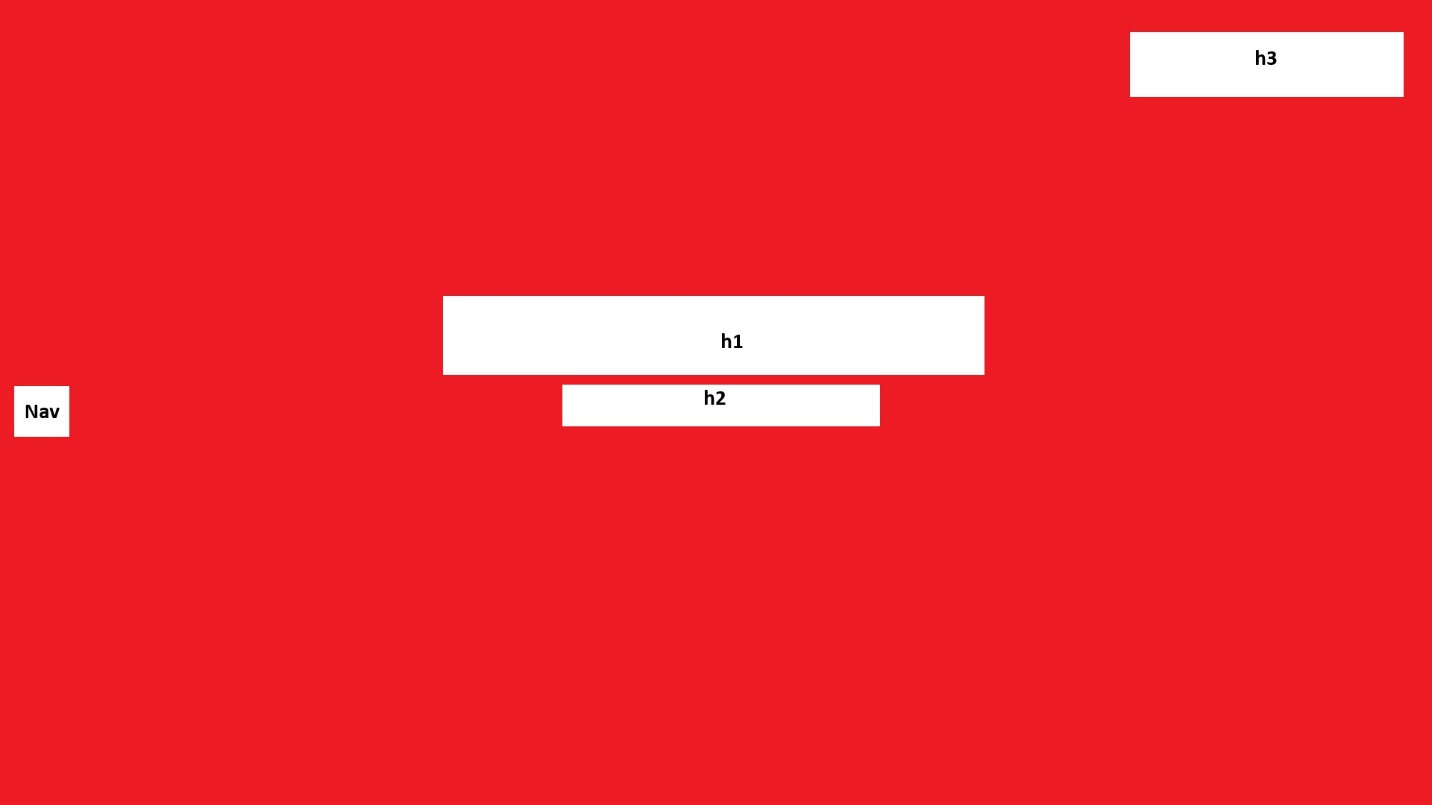
Automaticky generovaný popis

Drátené modely stránok

V tejto kapitole Vám podrobne predstavím a znázorním rozvrhnutie jednotlivých stránok.

Drátený model page\_landing.php

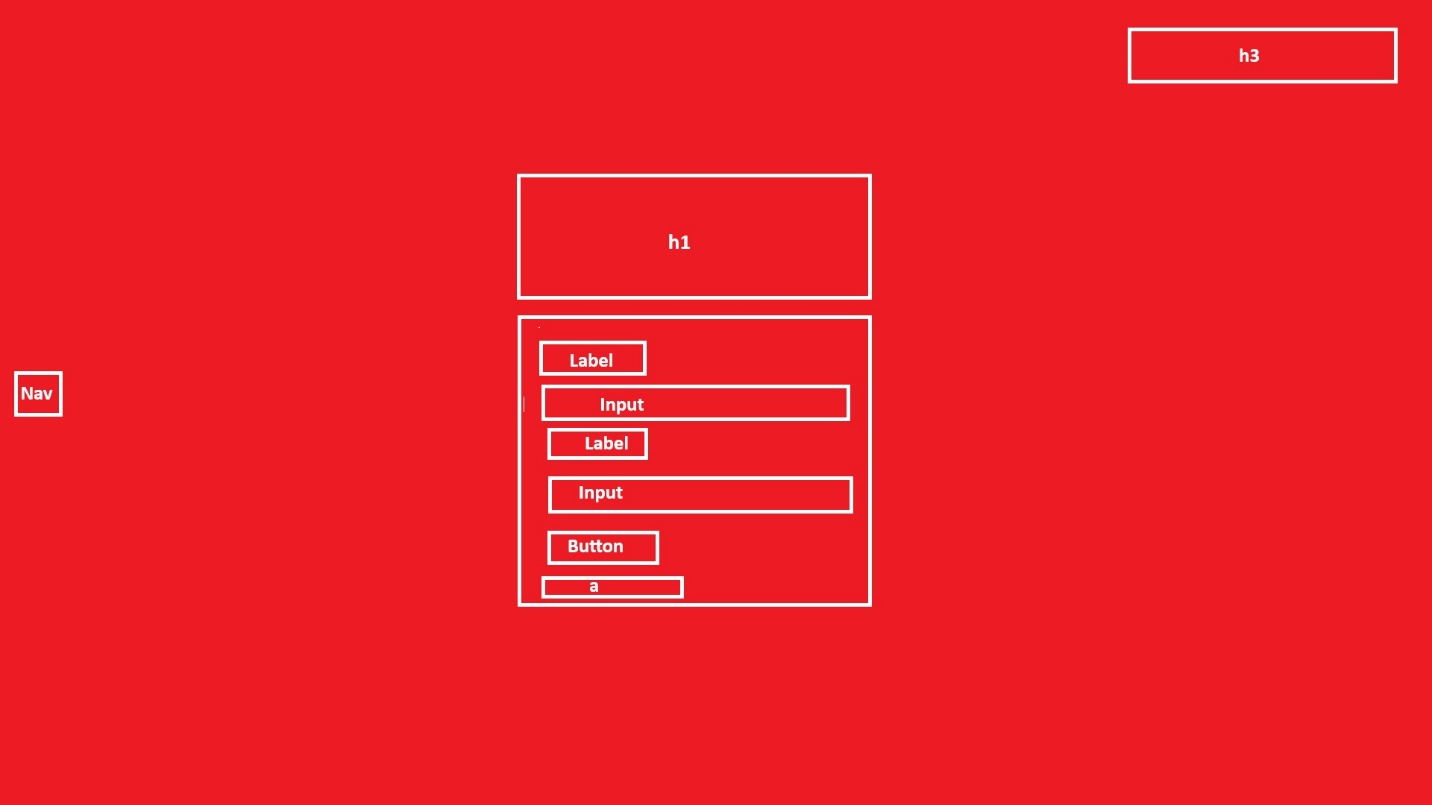
Úvodná stránka sa skladá z HTML elementov nav, h1, h2 a h3.



Obrázok 1: dráteny model page\_landing.php

Drátený model page\_login.php

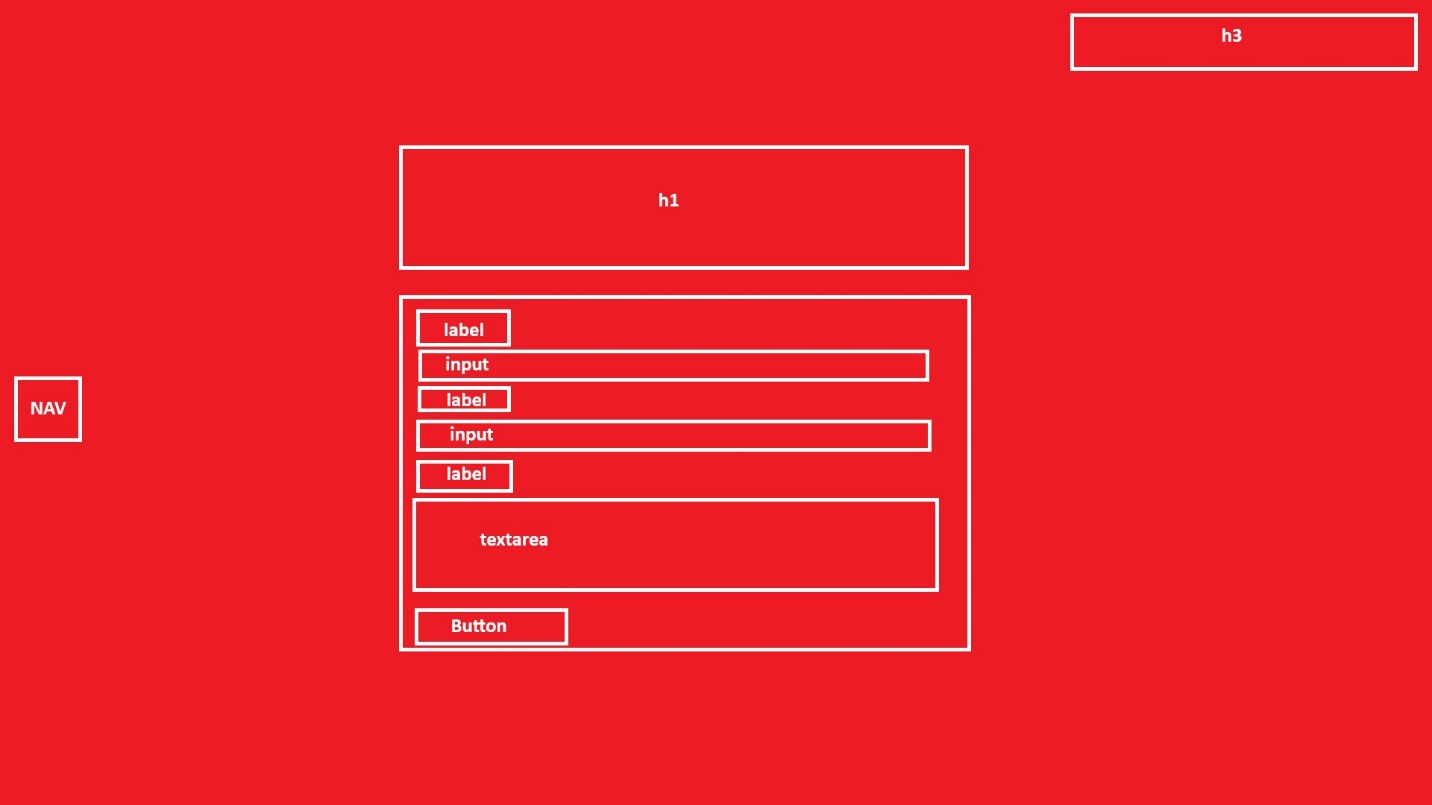
Stránka sa skladá z HTML elementov nav, h1, h3, label, input, button a anchor. Label a input elementy sú schované vo formulári.



Obrázok 2: drátený model page\_login.php

Drátený model page\_contactus.php

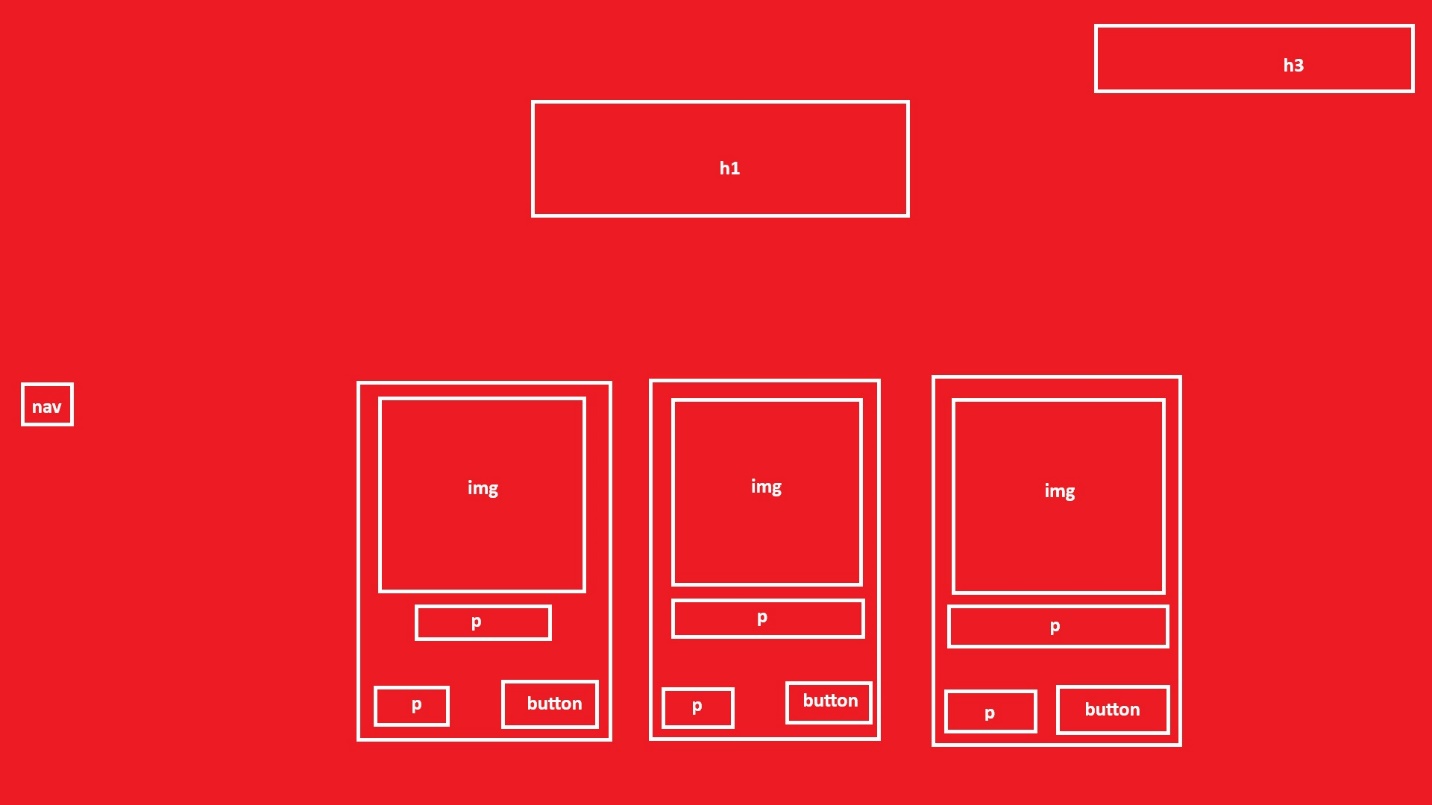
Stránka sa skladá z HTML elementov nav, h1, h3, label, input, textarea a button. Label, textarea a input elementy sú schované vo formulári.



Obrázok 3: Drátený model page\_contactus.php

Drátený model page\_food.php

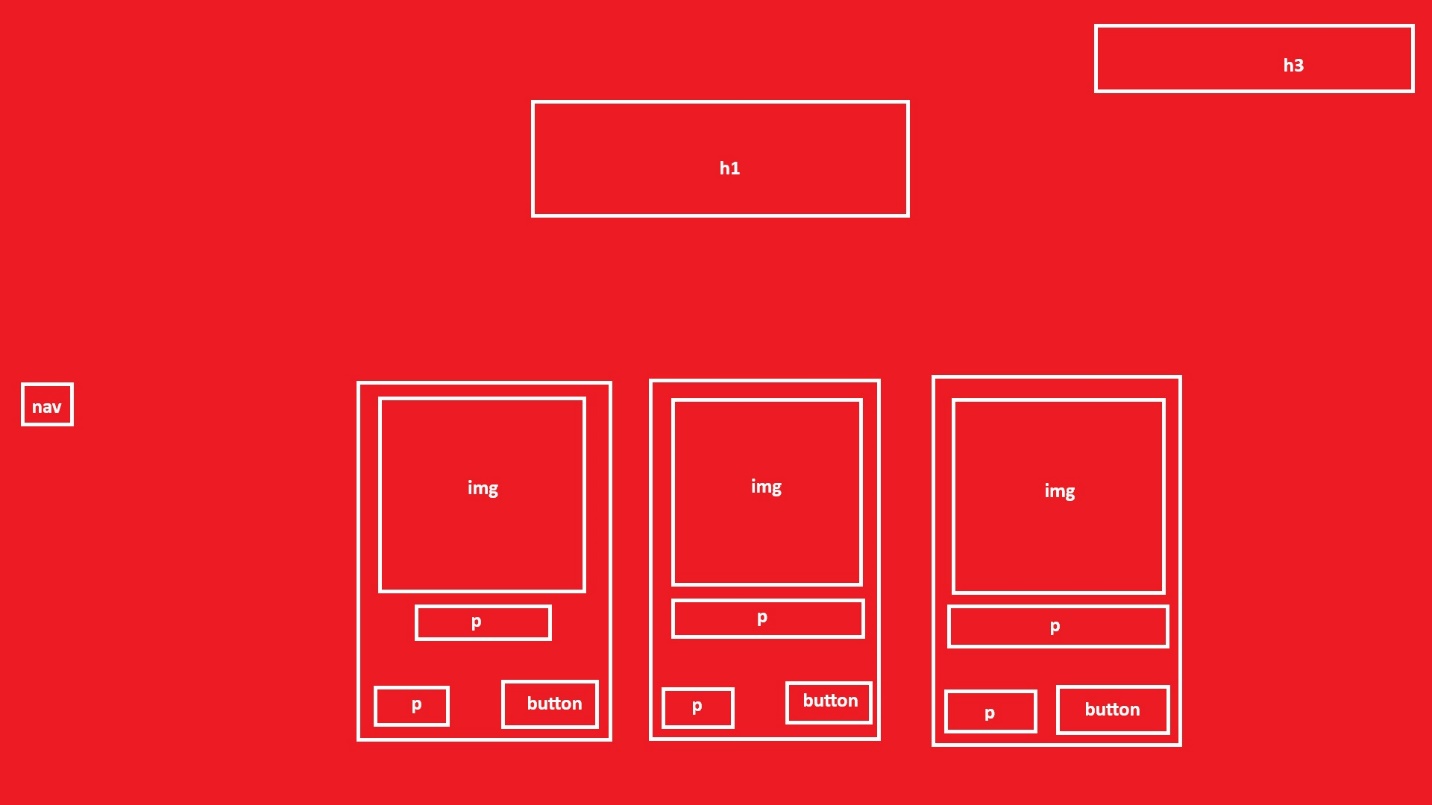
Stránka sa skladá z HTML elementov nav, h1, h3, label, input, textarea a button. Label, textarea a input elementy sú schované vo formulári.



Obrázok 4: Drátový model page\_food.php

Drátený model page\_suplements.php

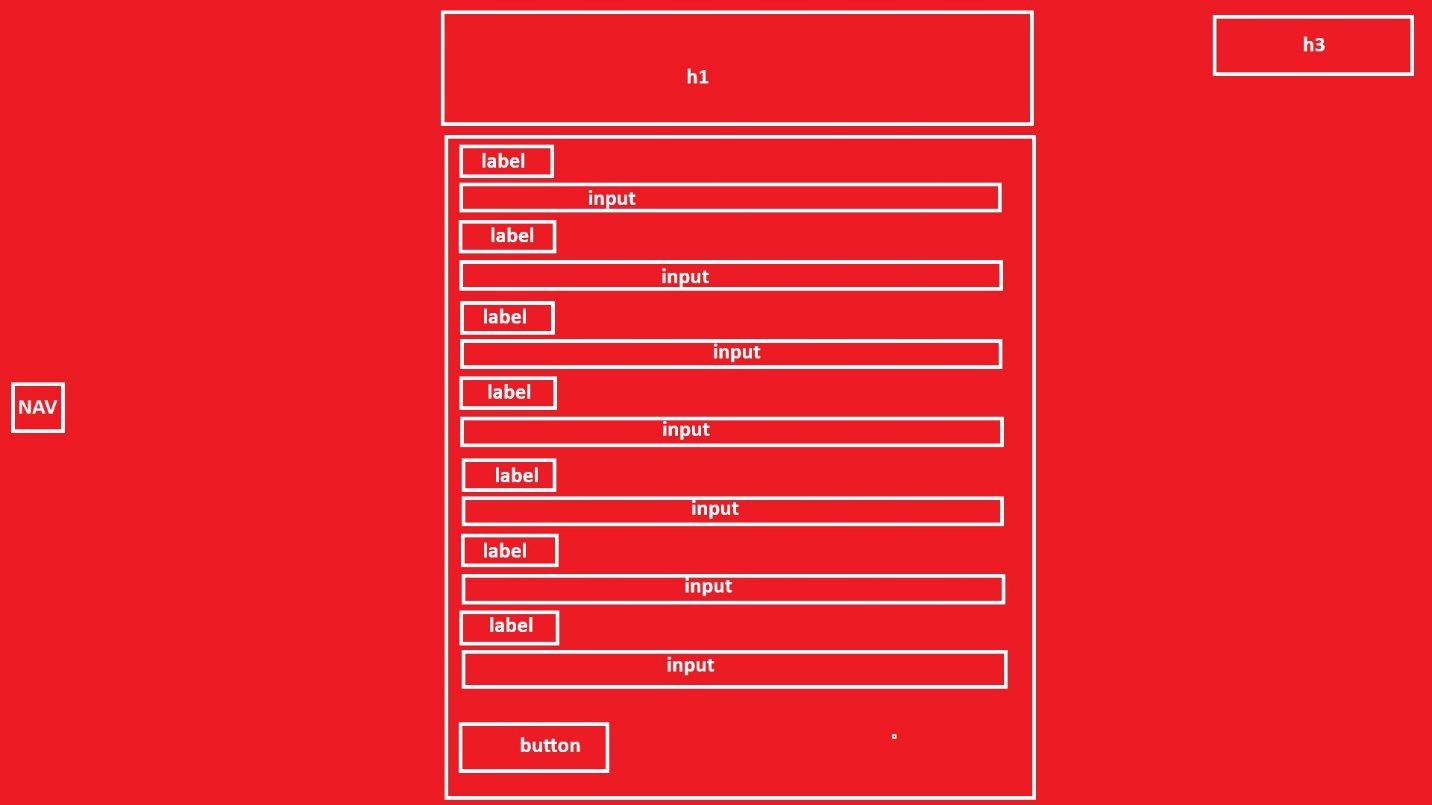
Stránka sa skladá z HTML elementov nav, h1, h3, label, input, textarea a button. Label, textarea a input elementy sú schované vo formulári.



Obrázok 5: Drátový model page\_suplements.php

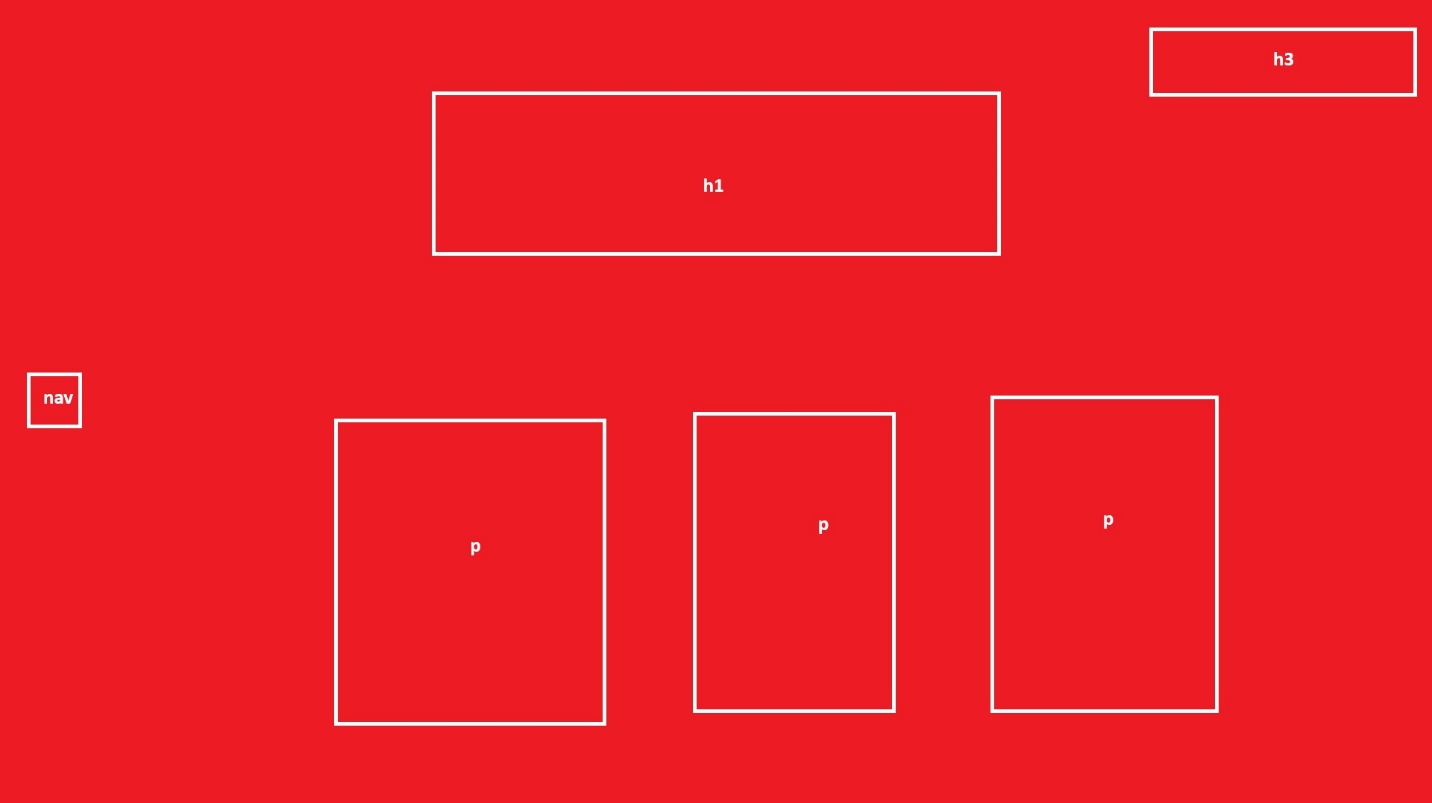
Drátený model page\_register.php

Stránka sa skladá z HTML elementov nav, h1, h3, label, input a button. Label a input elementy sú schované vo formulári.



Obrázok 6: Drátový model page\_register.php

Drátený model page\_admin.php

Stránka sa skladá z HTML elementov nav, h1, h3 a p. 

Obrázok 7: Drátený model pre page\_admin.php

Funkcie

Všetky funkcie ktoré mohli byť vybrané zo súborov sú uložene v súbori functions.php ktorý je následne používaný na stránkach pomocou include.

SendMessage($conn)

Funkcia sendMessage spracováva prichádzajúcu POST požiadavku z formulára a ukladá odoslané údaje do databázy.

Parametre:

$conn: Objekt pripojenia k databáze (MySQLi).

Popis:

Funkcia sendMessage vykonáva nasledujúce kroky:

Skontroluje, či je metóda požiadavky POST.

Načíta údaje z formulára, konkrétne name, email a message.

Získava aktuálny dátum a čas pomocou funkcie date.

Vloží údaje do tabuľky kontakt v databáze pomocou SQL dotazu.

Ak je vloženie úspešné, uzavrie pripojenie k databáze a zobrazí informačnú správu "Uspesne odeslano".

Ak sa vyskytne chyba pri vkladaní údajov, zobrazí chybovú správu s detailmi chyby.

loginUser($conn)

Funkcia loginUser spracováva prichádzajúcu POST požiadavku z formulára a autentifikuje používateľa na základe poskytnutých prihlasovacích údajov.

Parametre:

$conn: Objekt pripojenia k databáze (MySQLi).

Popis:

Funkcia loginUser vykonáva nasledujúce kroky:

Skontroluje, či je metóda požiadavky POST.

Načíta údaje z formulára, konkrétne email a password.

Heslo zašifruje pomocou funkcie md5.

Vykoná SQL dotaz, ktorý vyhľadáva používateľa v tabuľke zakaznik na základe emailovej adresy.

Ak sa nájde zodpovedajúci používateľ:

Skontroluje, či je heslo správne porovnaním zašifrovaného hesla s heslom v databáze.

Ak je heslo správne:

Spustí sa relácia pomocou session\_start.

Uložia sa údaje o používateľovi do relácie.

Zobrazí sa informačná správa o úspešnom prihlásení.

Presmeruje používateľa na stránku page\_landing.php.

Ak heslo nie je správne:

Zobrazí sa chybová správa o nesprávnom hesle.

Ak používateľ s daným emailom nie je nájdený, nezobrazí sa žiadna správa.

logoutUser()

Funkcia logoutUser ukončuje aktuálnu reláciu používateľa, odhlasuje ho a čistí všetky relácie.

Parametre:

Funkcia logoutUser nemá žiadne parametre.

Popis:

Funkcia logoutUser vykonáva nasledujúce kroky:

Spustí reláciu pomocou session\_start.

Vymaže všetky premenné relácie pomocou session\_unset.

Ukončí reláciu a vymaže všetky súbory relácie pomocou session\_destroy.

registerUser($conn)

Funkcia registerUser spracováva prichádzajúcu POST požiadavku z registračného formulára a registruje nového používateľa do databázy.

Parametre:

$conn: Objekt pripojenia k databáze (MySQLi).

Popis:

Funkcia registerUser vykonáva nasledujúce kroky:

Skontroluje, či je metóda požiadavky POST.

Načíta údaje z formulára: name, email, password, password2, city, address a telephone.

Heslo zašifruje pomocou funkcie md5.

Skontroluje, či sa heslá zhodujú:

Ak nie, zobrazí chybovú správu "Heslá sa nezhodujú".

Skontroluje, či užívateľ s daným emailom alebo telefónnym číslom už existuje v databáze:

Ak áno, zobrazí chybovú správu "Užívateľ s týmto emailom alebo číslom už existuje".

Ak všetky kontroly prejdú, vloží nového používateľa do tabuľky zakaznik v databáze.

Ak je vloženie úspešné, uzavrie pripojenie k databáze a zobrazí informačnú správu "Úspešne zaregistrovaný".

Ak sa vyskytne chyba pri vkladaní údajov, zobrazí chybovú správu s detailmi chyby.

getSuplements($conn)

Funkcia getSuplements načítava a zobrazuje doplnky stravy z databázy, ktoré sú dostupné na sklade.

Parametre:

$conn: Objekt pripojenia k databáze (MySQLi).

Popis:

Funkcia getSuplements vykonáva nasledujúce kroky:

Vykoná SQL dotaz na získanie všetkých položiek z tabuľky zbozi, kde druhZbozi je 2 (doplnky stravy).

Inicializuje pole $obrazky s cestami k obrázkom produktov.

Ak sú vrátené nejaké záznamy:

Pre každý záznam skontroluje, či je počet kusov (PocetKusov) väčší ako 0.

Ak áno, zobrazí produkt s jeho obrázkom, názvom, cenou a formulárom na pridanie do košíka.

Ak nie, pokračuje na ďalší záznam.

Ak nie sú vrátené žiadne záznamy, zobrazí správu "0 Výsledkov".

Uzavrie pripojenie k databáze.

getFoods($conn)

Funkcia getFoods načítava a zobrazuje potraviny z databázy, ktoré sú dostupné na sklade.

Parametre:

$conn: Objekt pripojenia k databáze (MySQLi).

Popis:

Funkcia getFoods vykonáva nasledujúce kroky:

Vykoná SQL dotaz na získanie všetkých položiek z tabuľky zbozi, kde druhZbozi je 1 (potraviny).

Inicializuje pole $obrazky s cestami k obrázkom produktov.

Ak sú vrátené nejaké záznamy:

Pre každý záznam skontroluje, či je počet kusov (PocetKusov) väčší ako 0.

Ak áno, zobrazí produkt s jeho obrázkom, názvom, cenou a formulárom na pridanie do košíka.

Ak nie, pokračuje na ďalší záznam.

Ak nie sú vrátené žiadne záznamy, zobrazí správu "0 Výsledkov".

Uzavrie pripojenie k databáze.

getAllOrders($conn)

Funkcia getAllOrders načítava a zobrazuje všetky objednávky z databázy, vrátane detailov o produktoch a zákazníkoch.

Parametre:

$conn: Objekt pripojenia k databáze (MySQLi).

Popis:

Funkcia getAllOrders vykonáva nasledujúce kroky:

Vykoná SQL dotaz na získanie všetkých objednávok (Objednavky), vrátane informácií o produktoch (Zbozi) a zákazníkoch (Zakaznik), pomocou INNER JOIN.

Inicializuje HTML kontajner pre zobrazenie objednávok.

Ak sú vrátené nejaké záznamy:

Pre každý záznam zobrazí detaily objednávky, vrátane ID objednávky, dátumu objednávky, ceny, ID produktu, názvu produktu a informácií o zákazníkovi.

Ak nie sú vrátené žiadne záznamy, zobrazí správu "0 Výsledkov".

Uzavrie pripojenie k databáze.

Komponenty

Component\_hlava.php

Začne HTML tag stránky a prepojí stránku s CSS a JS súborom.

Component\_nav.php

Obsahuje celý kód pre vysúvaciu navigáciu.

Database\_connect.php

Obsahuje funkciu pre vytvorenie spojenia s databázou.

Databáza

Relačný model MySQL databáze

Obrázok, na ktorom je text, snímka obrazovky, písmo, číslo

Automaticky generovaný popis

Obrázok 8: Relačný model databáze

Tento relačný model databázy predstavuje systém pre správu objednávok. Zahŕňa 5 tabuliek:

Databáza zakaznik

idZakaznik: Primárny kľúč, identifikuje zákazníka.

Meno: Meno zákazníka.

Email: Emailová adresa zákazníka.

Heslo: Heslo zákazníka.

Mesto: Mesto, kde zákazník žije.

Adresa: Adresa zákazníka.

TelefonneCislo: Telefonné číslo zákazníka.

Databáza Objednavky

idObjednavky: Primárny kľúč, identifikuje objednávku.

idZakaznik: Cudzí kľúč, odkazuje na zákazníka, ktorý objednávku vyrobil.

idZbozi: Cudzí kľúč, odkazuje na produkt, ktorý je v objednávke.

cena: Cena produktu v objednávke.

DatumObjednavky: Dátum vytvorenia objednávky.

Databáza zbozi

idZbozi: Primárny kľúč, identifikuje produkt.

Nazov: Názov produktu.

KodProduktu: Kód produktu.

PocetKusov: Počet kusov produktu v objednávke.

DruhZbozi: Cudzí kľúč, odkazuje na druh produktu.

Cena: Cena produktu.

Databáza druhyzbozi

idDruhy: Primárny kľúč, identifikuje druh produktu.

Druh: Popis druhu produktu.

Databáza kontakt

idKontakt: Primárny kľúč, identifikuje kontaktnú osobu.

Meno: Meno kontaktnej osoby.

Datum: Dátum kontaktu.

Email: Emailová adresa kontaktnej osoby.

Zprava: Správa z kontaktného formulára.

Vzťahy medzi tabuľkami

Zakaznik 1:N Objednavky: Jeden zákazník môže vytvoriť viacero objednávok.

Zbozi 1:N Objednavky: Jeden produkt môže byť v viacerých objednávkach.

Druhyzbozi 1:N Zbozi: Jeden druh produktu môže mať viacero produktov.

Použitie databázového modelu

Tento relačný model databázy môže byť použitý na:

Správu zákazníkov a ich objednávok.

Správu produktov a ich druhov.

Správu kontaktných formulárov.

JavaScript

JavaScript bol použitý pre menenie štýlou prvku arrowButton na otváranie a zatvárania nav komponentu, teda na jeho interaktivitu.

Main\_script.js

navOpen()

Popis: Otvorí navigačné menu.

Akcie:

Pridá triedu activeNav k elementu s ID nav.

Pridá triedu activeArrow k elementu s ID navArrowButton.

Nastaví atribút onclick na volanie funkcie navClose(), keď je navigačné menu aktívne.

navClose()

Popis: Zatvorí navigačné menu.

Akcie:

Odstráni triedu activeNav z elementu s ID nav.

Odstráni triedu activeArrow z elementu s ID navArrowButton.

Nastaví atribút onclick na volanie funkcie navOpen(), keď je navigačné menu neaktívne.

kosikOpen()

Popis: Otvorí nákupný košík.

Akcie:

Pridá triedu activeKosik k elementu s ID kosik.

Pridá triedu activeKosikButton k elementu s ID kosikButton.

Nastaví atribút onclick na volanie funkcie kosikClose(), keď je nákupný košík aktívny.

kosikClose()

Popis: Zatvorí nákupný košík.

Akcie:

Odstráni triedu activeKosik z elementu s ID kosik.

Odstráni triedu activeKosikButton z elementu s ID kosikButton.

Nastaví atribút onclick na volanie funkcie kosikOpen(), keď je nákupný košík neaktívny.

CSS

Styles.css

O rozmiestnenie HTML elementov sa mi stará vanilla CSS. Pre CSS všetkých stránok sa mi stará jeden CSS súbor. Neboli použitý žiaden externý framework.

Fotky

Na všetky fotky použité v práci a na stránke boli vytvorené a upravené mnou. Na stránku sa zobrazujú zo súboru fotky\_upravené. Originálne fotky sa nachádzajú v súbore fotky\_neupravené.

Záver

Zo svojej semestrálnej práce som celkom spokojný až na to že sa mi nepodarilo sfunkčniť košík do ktorého by si používateľ nahádzal všetky produkty ktoré chce a následne pod jednou objednávkou si ich objednal. Namiesto toho si musí jednotlivé produkty objednať zvlášť.

Účty:

Bežný:

[tomiskoller@gmail.com](mailto:tomiskoller@gmail.com)

heslo

Admin:

[admin@admin.cz](mailto:admin@admin.cz)

admin