

Gymnázium, Praha 6, Arabská 14

Programování

Ročníková práce



Duben
2024

Gabriela Stamenova, Jakub Vagera
Michael Vakula

Gymnázium, Praha 6, Arabská 14

Arabská 14, Praha 6, 160 00

Ročníková práce

Předmět: Programování

Téma: Plánování směn

Školní rok: 2023/2024

Autoři: Gabriela Stamenova, Jakub Vagera, Michael Vakula

Třída: 3.E

Vedoucí práce: Mgr. Jan Lána

Třídní učitel: Mgr. Blanka Hniličková

Poděkování

Na tomto místě bychom rádi poděkovali panu Petru Vagerovi za poskytnutí serveru, což nám umožnilo zveřejnění webové stránky. Také bychom chtěli poděkovat Mgr. Janu Lánovi za vedení práce.

Čestné prohlášení

Prohlašujeme, že jsem jedinými autoři tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V..... dne.....

Podpisy:

Anotace

Cílem práce je vytvořit webovou aplikaci, která by sloužila jako přehledné a efektivní prostředí pro organizaci směn. Nový uživatel si založí profil a zvolí si, zda chce vytvořit novou organizaci nebo připojit se do již existující. Manažer může pozvat nové členy do organizace prostřednictvím pozvánky. Dále si může vybrat, kolik směn chce mít v určitý den v daném týdnu či měsíci, a algoritmus mu je vygeneruje do organizace s časovou dobou délky směny. Člen organizace si pak může vybrat směny, které se mu hodí a následně se mu přesunou do jeho osobního kalendáře. Mezi členy bude funkce chatovací místnosti pro komunikaci. Celý program bude zprovozněn ve formě webu.

Abstract

The aim of this work is to create a web application, which would serve as a clear and efficient environment for organizing shifts. A new user creates a profile and chooses, whether to make a new organization or to join an already existing one. The manager can invite new members to the organization via invitations. They can also choose how many shifts they want to have on a certain day in a given week or month and an algorithm generates the shifts into the organization with their time periods. Employees can then choose the shifts they want and they will go into their personal calendar. There is a chatting room for better communication between employees. The entire program will be launched in the form of a website.

Obsah

1	Úvod	5
1.1	Zadání	5
2	Použité technologie	6
3	Struktura programu	7
3.1	Login	7
3.2	Kalendář	8
3.3	Vyhledávací systém	9
3.4	Vybírání směn	9
3.5	Databáze	10
3.6	Hlavní stránka admina	10
4	Objektová struktura	12
5	Struktura databáze	13
6	Závěr	14
7	Seznam obrázků	15
8	Zdroje	16
8.1	Informační zdroje	16
8.2	Zdroje obrázků	16

1 Úvod

Naše webová aplikace přináší jednoduché a efektivní řešení pro plánování pracovních směn v různých pracovních prostředích. Nabízí uživatelům širokou škálu nástrojů a funkcí pro správu směnových plánů, bez ohledu na velikost týmu nebo složitost organizační struktury. Každý uživatel má svou vlastní roli - může být administrátor, manažer nebo zaměstnanec. Každá role má přístup k určitým funkcím a možnostem. Administrátor může upravovat osobní informace zaměstnanců, vytvářet profily pro manažery a zaměstnance, modifikovat algoritmy a spravovat seznam validních IP adres pro přihlášení. Manažer může přidávat a odebírat směny zaměstnanců a přidávat nové pracovní objekty do systému. Zaměstnanci mají možnost samostatně plánovat své směny, které se automaticky zobrazí v jejich kalendáři. Pokud to povoluje jejich pracovní pozice, mohou si volit časové úseky a dny, ve kterých budou pracovat. Dále mají zaměstnanci možnost komunikovat mezi sebou pomocí chatovacího prostoru, což napomáhá efektivní spolupráci v týmu.

1.1 Zadání

Zadání naší ročníkové práce bylo:

Vytvořit webovou aplikaci, která umožňuje snadné a efektivní plánování směn v pracovním prostředí. Naše aplikace umožňuje snadno vytvářet, upravovat a sdílet směnové plány se zaměstnanci. Každý uživatel se přihlásí pod svoji příslušnou roli - admin, manažer nebo zaměstnanec. Každá role má určité funkce a možnosti. Admin může měnit osobní informace zaměstnanců, vytvářet profily pro manažery i zaměstnance, modifikovat algoritmy a měnit IP adresy na seznamu validních IP adres pro přihlášení. Manažer může přidávat a odebírat směny zaměstnanců a přidávat nové objekty do společnosti. Zaměstnanec si může sám plánovat směny, které se mu poté přidají do kalendáře. Může si sám určovat časovou dobu a dny, ve které bude pracovat, pokud to pracovní pozice umožňuje. Zaměstnanci mají také možnost komunikovat mezi sebou pomocí chatovací místnosti.

2 Použité technologie

K vyhotovení aplikace jsme použili integrované vývojové prostředí Visual Studio Code, což je editor zdrojového kódu vyvíjený společností Microsoft pro operační systémy Windows, Linux a macOS. Pro programování samotné aplikace jsme zvolili programovací jazyk PHP, jelikož je především určený k programování dynamických internetových stránek a webových aplikací nejčastěji ve formátu HTML a XHTML. To mě vede k tomu, že jsme pro design stránky použili jazyky HTML a CSS a pro spravování kalendáře JavaScript. Dále jsme využili phpMyAdmin - nástroj napsaný v jazyce PHP umožňující jednoduchou správu obsahu databáze MySQL prostřednictvím webového rozhraní. Pro řízení databází jsme použili právě MySQL (My Structured Query Language neboli systém pro řízení databází). Jedná se o multiplatformní databázi, která komunikuje prostřednictvím jazyka SQL. Běžně se využívá v kombinaci s PHP a phpMyAdmin. Dále jsme použili AJAX, což je obecné označení pro technologie vývoje interaktivních webových aplikací, které mění obsah svých stránek bez nutnosti jejich kompletního znovunačítání. K vypracování dokumentace jsme použili LaTeX Overleaf.



Obr. 1 - Loga použitých nástrojů

3 Struktura programu

Jak bylo již zmíněno, program je rozdělen do tří hlavních částí podle toho, jestli se uživatel přihlašuje jako admin, manažer nebo zaměstnanec. V této kapitole budou podrobněji popsány funkce a principy programu, které jsou pro všechny uživatele stejné.

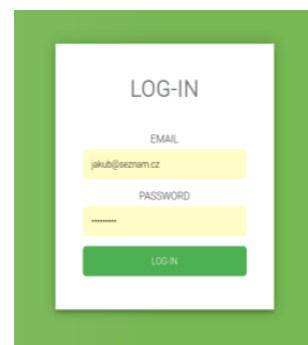
3.1 Login

První část naší webové stránky je kód na přihlášení do aplikace *Login*. K jeho vyhotovení jsme použili jazyky PHP, HTML a Javascript, kde každý z nich plní určitou funkci.

PHP část: Kód začíná blokem PHP, který nastavuje proměnnou `is_invalid` na `false`. Tato proměnná slouží k označení, zda byl přihlašovací pokus neúspěšný. Pokud je žádost odeslána metodou POST (způsob odeslání dat z formuláře na server), začíná se provádět proces ověřování přihlašovacích údajů. Je navázáno spojení s databází a provede se na ni dotaz pro ověření existence uživatele s daným e-mailem. Pokud je uživatel nalezen, ověří se heslo pomocí funkce `password_verify` a spustí se funkce `session`. Ta umožňuje udržovat stav přihlášeného uživatele mezi různými stránkami aplikace a poskytuje způsob, jak ukládat a získávat data o uživateli mezi jednotlivými HTTP požadavky. Poté je uživatel přesměrován na vhodnou stránku podle své role (admin, manažer, zaměstnanec) pomocí funkce `header`. Ta je vyvolána ve chvíli, kdy jsou přihlašovací údaje ověřeny a uživatel je úspěšně přihlášen. Podle role uživatele se provede přesměrování na odpovídající stránku.

HTML část: Následuje kód HTML pro samotnou stránku s formulářem pro přihlášení. V HTML formuláři jsou pole pro zadání e-mailu a hesla. Pokud došlo k neúspěšnému přihlášení (proměnná `is_invalid` je `true`), zobrazí se upozornění o neplatném přihlášení.

JavaScript část: Obsahuje skript, který vypne automatické vyplňování hesla v prohlížeči. To zajistí ochranu senzitivních údajů jako je právě heslo.

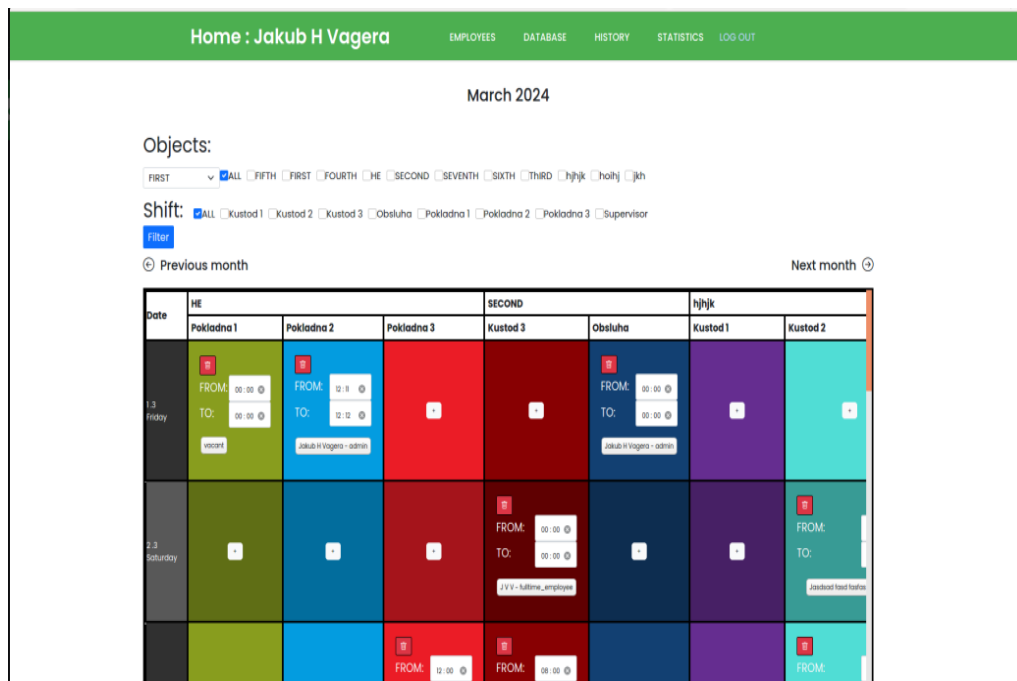


Obr. 2 - Login

3.2 Kalendář

Každý uživatel má přístup ke společnému kalendáři, kam se zapisují směny zaměstnanců. Kalendář slouží k přehledné organizaci směn a k řádnému fungování firmy. Hlavní kód pro kalendář se nachází ve třídě `calendar.php`. Na začátku dochází k volání `session_start()`, což je funkce pro spuštění session na serveru. Ta slouží k ukládání informací o uživateli na serveru a umožňuje jejich použití v různých částech webové aplikace. Poté následuje podmínka `if`, která kontroluje, zda existuje proměnná v session s názvem `user2_id`. Jestliže ano, provádí se několik dalších operací.

Nejprve se sestavuje SQL dotaz pro vybrání uživatele z databáze tabulky `user2`, kde ID odpovídá ID uloženému v session. Poté se provádí dotaz na databázi s použitím objektu `mysqli`, který je již inicializován pomocí souboru `database.php`. Výsledek tohoto dotazu se ukládá do proměnné `$result`. Z tohoto výsledku se načítá řádek (pokud existuje) do pole `$user` pomocí `fetch_assoc()` metody. Dále se provádí další SQL dotaz pro získání pozice a ID uživatele a výsledek se ukládá do proměnných `$userp` a `$userid` ve while smyčce. Poté následuje HTML a CSS část dokumentu. Tato část obsahuje odkazy na externí zdroje stylů (Bootstrap, ikony atd.) a také vlastní CSS styly pro úpravu vzhledu webové stránky kalendáře.



Obr. 3 - Ukázka kalendáře

3.3 Vyhledávací systém

Náš vyhledávací systém funguje na základě SQL databáze. Jak jsem již zmínila, při vytvoření profilu uživatele, jsou jeho osobní údaje ukládány do databáze. Admin a manažer mají možnost vyhledávat zaměstnance podle jména, příjmení, popř. rodného příjmení. Když uživatel zadá pouze iniciály jmen a příjmení zaměstnance, tak vyhledávací systém najde všechny odpovídající kombinace, které jsou v databázi.

Kód začíná s připojením k databázi pomocí objektu `mysqli`. Předpokládá se, že konfigurace databáze je uložena v souboru `database.php`. Dále probíhá zpracování vstupních dat z formuláře pomocí metody `POST`. Zadávaná data se rozdělí na pole pomocí mezery a poté se použijí k vyhledání v databázi. Následně na základě vstupních dat se sestavuje SQL dotaz, který vyhledá uživatele podle jména a příjmení. Výsledky vyhledávání se zpracovávají a vypisují do HTML stránky. Pro každého nalezeného uživatele se zobrazují informace jako jméno, příjmení, e-mail, telefon, pozice a přiřazené směny. Celkově tato funkce umožňuje dynamicky vyhledávat uživatele v databázi a zobrazovat jejich informace v přehledné podobě na webové stránce.

3.4 Vybírání směn

Směny se zaměstnancům přiřazují na základě jejich požadavků a časových možností. Algoritmus vyhodnocuje požadovanou směnu a možnosti zaměstnanců, a poté ji přiřadí tomu nejvhodnějšímu. Algoritmus funguje na následujícím principu: najde všechny uživatele, kteří mohou v daný den pracovat, včetně toho, zda jsou na plném nebo polovičním úvazku. Dále upřednostňuje zaměstnance, kteří mají méně směn, a kontroluje časové intervaly, aby se jim požadovaná směna nepřekrývala s jinou. Potom přichází tzv. Back track algoritmus, který slouží k dosažení nejvyšší efektivity. Např. když při prvním načtení vybere Z1 mezi Z1 a Z2, ale poté zjistí, že Z1 má jinou směnu, tak se vrátí zpátky, provede nové vyhledávání a vybere Z2 v tomto případě. Zaměstnanci jsou si navzájem rovni a při větším zájmu je vybírání je náhodné.

3.5 Databáze

Třída `Database.php` slouží k navázání spojení s databází pomocí rozhraní `MySQLi`. Na začátek se definují proměnné obsahující informace potřebné pro připojení k databázi. Mezi tyto informace patří `hostname` (`$host`), název databáze (`$dbname`), uživatelské jméno (`$username`), heslo (`$password`) a volitelně port (`$port`). V kódu jsou připraveny dvě sady parametrů pro připojení k různým databázím, přičemž jedna je pro lokální databázi a druhá pro globální databázi. V další části je vytvořeno spojení s databází pomocí konstruktoru třídy `mysqli`. Vytvořené spojení je uloženo do proměnné `$mysqli`. V případě, že připojení k databázi selže, je zobrazena chybová zpráva, která obsahuje informaci o chybě spojení. Nakonec je proměnná `$mysqli` vrácena jako výsledek volání funkce. Tato proměnná obsahuje objekt připojení k databázi a může být použita k provádění dotazů a manipulaci s daty v databázi.

3.6 Hlavní stránka admina

Kód pro hlavní stránku admina se nachází ve třídě `admin_main_page.php`. V této podkapitole budou popsány jednotlivé části kódu, které umožňují připojení administrátorské stránky k databázi.

Otevření spojení s databází: Skript začíná otevřením session (`session_start()`) a kontrolou, zda je uživatel přihlášen. Pokud je uživatel přihlášen (jeho identifikátor je uložen v `$_SESSION["user2_id"]`) a pokračuje se v kódu.

Připojení k databázi: Kód používá funkci `require` k načtení souboru `database.php`, který obsahuje nastavení pro připojení k databázi. Poté se provádí SQL dotaz pro získání informací o přihlášeném uživateli.

Vytvoření navigačního panelu: HTML kód pro navigační panel obsahuje odkazy na různé stránky administrátorského rozhraní, jako jsou seznam zaměstnanců, databáze, historie, statistiky a možnost odhlášení.

Zobrazení informací o uživateli: Na stránce jsou zobrazeny informace o přihlášeném uživateli - jméno, příjmení, e-mail, telefonní číslo, heslo, a pozice.

Zobrazení nejbližších směn: Pro každý den (včera, dnes a zítra) se zobrazují informace o nejbližších plánovaných směnách.

Záznam docházky: Uživatel může potvrdit svůj příchod a odchod pomocí tlačítek na stránce. Tyto údaje se ukládají do databáze. Každý zaměstnanec má povinnost tato pole vyplňovat pravidelně, aby mohli manažeři a případně admin lépe dohlížet na přesnost ze strany zaměstnanců.

Záznamy docházky a komentáře: Zobrazují se informace o docházce a komentáře k jednotlivým směnám. V případě, že uživatel přichází pozdě nebo odchází dříve, musí napsat krátké odůvodnění.

Informační deska a aktuální směny: Na stránce se zobrazuje informační deska s aktuálními směnami a seznamem zaměstnanců. Ta slouží k přehledné orientaci v nadcházejících směnách.

Na stejném principu fungují i stránky manažera a zaměstnance, akorát každá z nich má příslušné práva a funkce.

4 Objektová struktura

Aby mohl admin předem vytvořit směnu, musí nejdříve vytvořit objekt na, kterém tato směna bude. Za tzv. objekt se považuje místo, kde bude směna vykonávána, například kancelář nebo pobočka. Admin má při vytváření objektu možnost, buďto vytvořit tzv. hlavní objekt nebo může vytvořit tzv. podobjekt. Hlavní objekty se v systému objevují odděleně a jejich hlavním účelem je snadnější zobrazení a přehled v systému. Podobjekt je objekt, který se váže na hlavní objekt či další podobjekty, jejichž hlavním cílem je vytvořit hierarchii v systému. V databázi jsou objekty uloženy v tabulce Objects.

Struktura je:

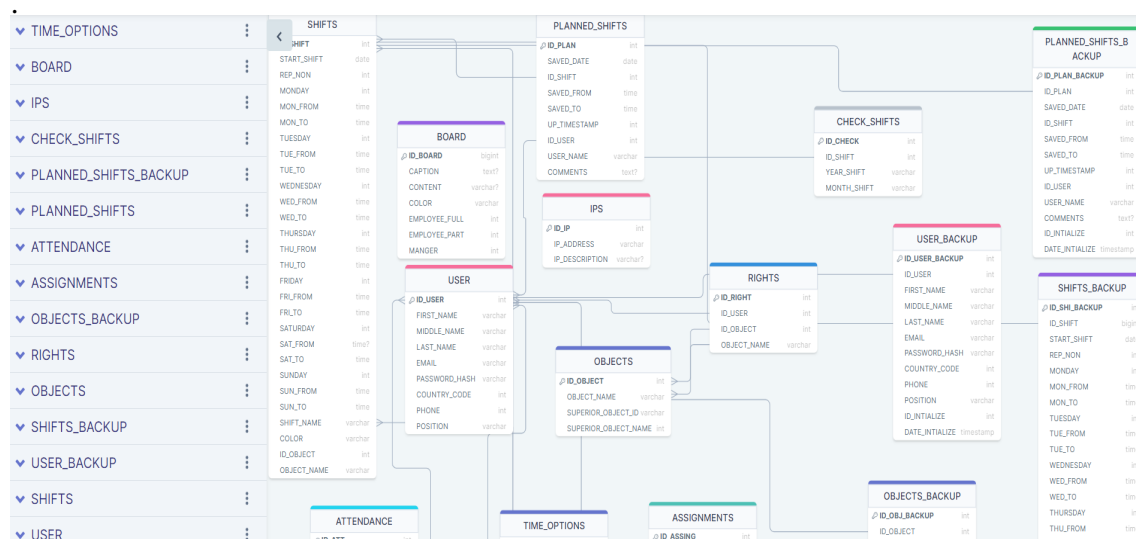
ID_OBJECT	NAME_OBJECT	SUPERIOR_OBJECT_ID	SUPERIOR_OBJECT_NAME
1	hlavní budova		
2	1. Patro	1	hlavní budova
3	2. Patro	1	hlavní budova

Obr. 4 - Ukázka objektové struktury

Hlavní objekty jsou v databázi ty objekty, které nemají `superior_object_id`, ty, které ho mají jsou podobjekty. Při vyhledávání nebo výpisu objektu používá program rekurzivní funkci. Tato funkce prochází všechny podobjekty daného objektu a zobrazuje je stejně jako hlavní objekty.

5 Struktura databáze

Databáze se skládá ze 16 tabulek. Každá z nich má svoji funkci. Názvy jsou samovysvětlující, tudíž není potřeba je příliš rozepisovat. Např. TIME_OPTIONS ukládá časové možnosti zaměstnanců, které se pak volají při přidělování směn. ATTENDANCE zaznamenává příchod a odchod zaměstnanců a případně důvod zpoždění. ASSIGNMENTS zodpovídá za požadavky zaměstnance typu: kde bude zrovna pracovat či jakou pozici bude plnit. Všechna data se ukládají do této SQL databáze a následně se volají po připojení s aplikací pomocí objektu mysqli.



Obr. 5 - Struktura databáze

6 Závěr

Výsledkem naší práce je webová aplikace pro plánování směn, určena například firemním organizacím. Vše se povedlo povedlo podle očekávání bez žádných až moc závažných komplikací. Nápad na budoucí zlepšení je vytvoření mobilní verze aplikace, což umožní zaměstnancům snadnější přístup. Práce nás naučila hodně nových věcí, týkajících se jak programování, tak týmového zpracování. Není snadné vytvářet komplexní aplikaci v týmu, ale při správném přístupu, řádné komunikaci a spravedlivém rozdělení práce nevznikají velké problémy. Výslednou aplikaci hodnotíme jako úspěšnou: nabízí uživatelům snadný a intuitivní způsob plánování a správy směn, což vede k lepší organizaci práce a zvýšení efektivity zaměstnanců.

7 Seznam obrázků

Obr. 1 - Loga použitých nástrojů [4]

Obr. 2 - Login [5]

Obr. 3 - Ukázka kalendáře [6]

Obr. 4 - Ukázka objektové struktury [10]

Obr. 5 - Struktura databáze [13]

8 Zdroje

8.1 Informační zdroje

1. Visual Studio Code [Online] [Citace: 15. 03. 2024] https://cs.wikipedia.org/wiki/Visual_Studio_Code
2. PhpMyAdmin [Online] [Citace: 15.03.2024] <https://www.phpmyadmin.net/>
3. MySQL [Online] [Citace 15.03.2024] <https://www.itnetwork.cz/mysql/mysql-tutorial-uvod-a-priprava-prostredi>
4. AJAX [Online] [Citace: 15.03.2024] [://cs.wikipedia.org/wiki/AJAX](https://cs.wikipedia.org/wiki/AJAX)

8.2 Zdroje obrázků

1. Visual Studio Code: https://www.google.com/search?q=visual+studio+code+logo&sca_esv=1df9984c226b3a13&udm=2&biw=1536&bih=695&ei=9FMnZtGLA5r-i-gPzvytmAI&ved=0ahUKEwjR1ZrG2deFAxUa_wIHHU5-CyMQ4dUDCBA&uact=5&oq=visual+studio+code+logo&gs_lp=Egxnd3Mtd2l6LXNlcjF3Zpc3VhbCBzdHVkaW8gY29kZSBsb2dvMgUQABiABDIEEAAYHjIEEAAYHjIEEAAYHjIEEAAYHjIEEAAYHjIEEAAYHjIEEAAYHjIGEAAYBRgeSKUCUJSHWN0acAJ4AJABAJgBbbqABuQSQAQM0LjK4AQPIAQD4AQGYAgagAQEqEwgIKEAAYgaQYQxiKBcICBhAAGAcYHpgDAIgGAZIHAzQuM6AHgxw&sclient=gws-wiz-serp&safe=active&ssui=on#vhid=PryLAM8_xQK-gM&vssid=mosaic
2. Latex Overleaf: <https://www.google.com/url?sa=i&url=https%3A%2F%2Fde.wikipedia.org%2Fwiki%2FOverleaf&psig=A0vVaw0Klf6nVSrPerAjA9nJHaAn&ust=1713939993773000&source=images&cd=vfe&opi=89978449&ved=0CBAQjRxqFwoTCNDs65fa14UDFQAAAAdAAAAABAZ>
3. PhpMyAdmin: https://www.google.com/url?sa=i&url=https%3A%2F%2Fen.m.wikipedia.org%2Fwiki%2FFile%3APhpMyAdmin_logo.svg&psig=A0vVaw0TdWtnu3oWvv0byO-9AMVP&ust=1713940116571000&source=images&cd=vfe&opi=89978449&ved=0CBAQjRxqFwoTCLDQs9Pa14UDFQAAAAAdAAAAABAE
4. HTML, CSS: <https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.vhv>

.rs%2Fviewpic%2FhJoiJoT_html-css-logo-png-transparent-png%2F&psig=A0vVaw2-xumZapI7H0576Yew3tAp&ust=1713940148719000&source=images&cd=vfe&opi=89978449&ved=0CBAQjRxqFwoTCIDrkeva14UDFQAAAAAdAAAAABAE

5. Javascript: https://www.google.com/url?sa=i&url=https%3A%2F%2Fcommons.wikimedia.org%2Fwiki%2Ffile%3AJavascript_badge.svg&psig=A0vVaw1FP0bCZbevTyQS07_WoUru&ust=1713940210534000&source=images&cd=vfe&opi=89978449&ved=0CBAQjRxqFwoTCOCVw4rb14UDFQAAAAAdAAAAABAE

6. MySQL: https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.logo.wine%2Flogo%2FMySQL&psig=A0vVaw15lT9_rXNC2nq0_AoItPAo&ust=1713940274605000&source=images&cd=vfe&opi=89978449&ved=0CBAQjRxqFwoTCMC79pzb14UDFQAAAAAdAAAAABAE