

Programátorská Dokumentace - Cookbook

Nástroje a Technologie

- **PHP:** Použitý na backend
- **JSON:** Ukládání dat o uživatelích (users.json) a receptech (prispevky.json).
- **HTML:** Struktura webových stránek.
- **CSS:** stylování webových stránek
- **JavaScript:** Klientská validace a interaktivní funkce (script.js), AJAX.
- **PHPDocumentor:** Automatická PHP dokumentace

Struktura Projektu

- **index.php:** Hlavní stránka zobrazující seznam receptů s možností filtrování a stránkování.
- **fce.php:** Obsahuje pomocné funkce pro práci s JSON soubory, hashování hesel a validaci dat.
- **registration.php:** Stránka pro registraci nových uživatelů s omezením délky uživatelského jména na 20 znaků.
- **login.php:** Stránka pro přihlášení uživatelů do systému.
- **logout.php:** Skript pro odhlášení uživatele a zničení jeho sessiony.
- **mojeprispevky.php:** Stránka pro zobrazení a správu vlastních receptů přihlášeného uživatele.
- **proajax.php:** zpracovává AJAX požadavky na kontrolu dostupnosti uživatelského jména
- **pridatprispevek.php:** Stránka pro přidání nového receptu do databáze.
- **vypisuzivatelu.php:** Administrativní stránka pro výpis a správu uživatelských účtů.
- **script.js:** JavaScript soubor obsahující funkce .
- **index2.css, form.css, mojeprispevky.css, profil.cc, vypisuzivatelu.cc:** CSS soubory pro stylování aplikace a formulářů. Obsahují styly pro responzivitu.
- **print.css:** CSS soubor pro přehledné zobrazení pro tisk
- **users.json:** JSON soubor ukládající informace o uživatelích.
- **prispevky.json:** JSON soubor ukládající informace o receptech.
- **obrazky:** Složka pro ukládání obrázků receptů, včetně výchozího obrázku default.png.

Funkce a Logika

Registrace a Přihlášení

- **Registrace:** Uživatelé mohou vytvořit účet vyplněním registračního formuláře, kde zadají uživatelské jméno (max. 20 znaků), heslo a jeho potvrzení. Serverová validace zajišťuje, že uživatelské jméno je unikátní, splňuje délková kritéria a hesla se shodují (AJAX). Hesla jsou bezpečně hashována před uložením do JSON souboru. Po úspěšné registraci je uživatel automaticky přihlášen.
- **Přihlášení:** Registrovaní uživatelé se mohou přihlásit zadáním svého uživatelského jména a hesla. Server ověřuje správnost zadaných údajů a po úspěšném přihlášení nastaví session proměnné pro sledování uživatelského stavu.

Filtrování a Stránkování Receptů

- **Filtrování:** Uživatelé mohou filtrovat recepty podle kategorií pomocí dropdown menu. Pomocí tlačítka recept, vrátit zpátky všechny příspěvky.
- **Stránkování:** Recepty jsou řazeny od nejnovějších a rozděleny do stránek po pěti receptech.

Admin funkce

- **Mazání Receptů:** Admini mají možnost mazat recepty přímo z hlavní stránky. Při mazání se odstraní nejen recept, ale i jeho přidružený obrázek, pokud není výchozí.
- **Správa Uživatelských Rolí:** Administrátoři mohou spravovat uživatelské účty, včetně změny rolí mezi user a admin, mohou ale pouze povýšit usera na admina. Tímto způsobem je zajištěna kontrola přístupových práv v rámci aplikace.

Datové Struktury

users.json

Tento soubor ukládá informace o všech registrovaných uživateli. Každý uživatel je reprezentován objektem obsahujícím:

```
{  
  "username": "user123",  
  "password": "hashovaneheslo123.....",  
  "role": "user"  
}
```

- **username:** Uživatelské jméno.
- **password:** Hashované heslo.
- **role:** user nebo admin

prispevky.json

Tento soubor uchovává data o všech receptech. Každý recept zahrnuje:

```
{  
  "id": "123",  
  "title": "Název receptu",  
  "content": "popis a postup"  
  "author": "username autora",  
  "category": "kategorie receptu",  
  "ingredients": "seznam položek",  
  "post_time": "datum nahrání",  
  "image_path": "obrazky\123 "  
}
```

Bezpečnostní Opatření

1. CSRF Ochrana:

- Všechny formuláře obsahují skrytý CSRF token, který je generován při každé session. Server ověřuje platnost tohoto tokenu před zpracováním požadavků.

2. Ochrana proti XSS:

- Všechny výstupy používají htmlspecialchars, což zabraňuje vkládání JavaScript kódu do aplikace a chrání uživatele před Cross-Site Scripting útoky.

3. Hashování Hesel:

- Hesla jsou hashována pomocí password_hash s algoritmem PASSWORD_DEFAULT před uložením do JSON souborů. Při přihlášení se používá password_verify pro ověření správnosti hesla bez jeho přímého ukládání.

4. Role:

- Uživatelé mají přiřazené role (user a admin), které určují jejich oprávnění v aplikaci.
- Nepřihlášení uživatelé: nejsou v databázi, vidí jen index.php (mohou si zobrazit nové recepty), login.php, registration.php.
- Uživatelé: přidávají příspěvky, upravují vlastní příspěvky, mají k nim přístup na vlastní stránce mojeprispevky.php
- Admini: přidávají příspěvky, upravují vlastní příspěvky, mají k nim přístup na vlastní stránce mojeprispevky.php mažou veškeré příspěvky, mění role uživatelů na adminy v vypisuzivatelu.php.