ZWA – Semistralní práce

Beliaev Daniil

**Zadání.**

Je nutné vyvinout web pro zaznamenávání a kontrolu denních úkolů. Web by měl mít možnost přidávat, mazat, označovat a upravovat úkoly. Na webu bude možné se zaregistrovat, autorizovat. Měla by být také přítomna možnost přepínání barevných motivů. Web musí být přizpůsoben libovolnému rozlišení obrazovky. Web bude vhodný pro lidi, kteří potřebují denně zaznamenávat velké množství úkolů. Je ideální pro studenty, učitele, podnikatele, manažery, programátory atd.

Uživatelské role:

1. Návštěvník (nepřihlášený uživatel)
2. Uživatel (po přihlášení)

Funkce:

1. Neregistrovaný uživatel vidí pouze uvítací stránku a má možnost se zaregistrovat na webu nebo změnit barevné téma.
2. Registrovaný uživatel se může přihlásit, zrušit autorizaci, pracovat s tudu listem (přidat, odstranit, označit, upravit úkoly)

Stack:

HTML, CSS, JavaScript, PHP, Databáze, MySQL

Akceptační podmínky:

**Úvodní stránka.** V horní části stránky (na všech stránkách) je logo se sloganem a navigační nabídka. V levé horní části stránky je přepínač barevných motivů. Existují dvě témata - tmavá a světlá, ve výchozím nastavení je světlá. Uprostřed stránky je uvítací text. Vpravo od textu je tlačítko Registrace.

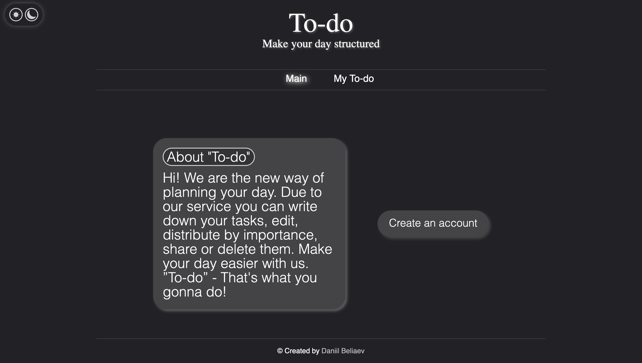
**Registrační stránka.** Uprostřed obrazovky je registrační okno s poli pro zadání nicknamu, e-mailu, hesla, opakování hesla. Každé pole má štítek. Oranžová hvězdička označuje, která pole jsou povinná. Níže je nápis vysvětlující označení hvězdičky. Na samém dně formuláře je tlačítko pro odeslání formuláře. Kliknutím na tlačítko se před uživatelem otevře autorizační okno.

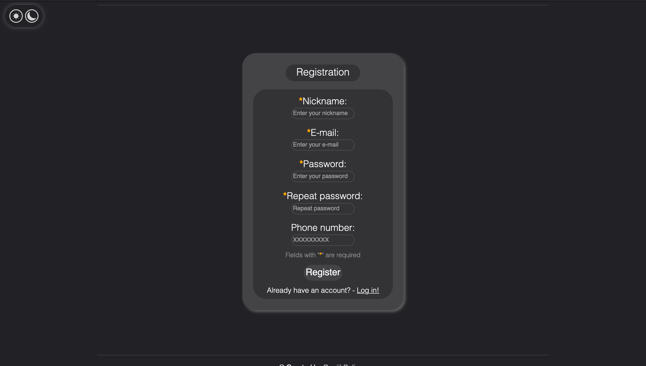
**Autorizační stránka.** Uprostřed obrazovky je autorizační okno s poli pro zadání e-mailu, hesla. Každé pole má štítek. Oranžová hvězdička označuje, která pole jsou povinná.

Níže je nápis vysvětlující označení hvězdičky. Na samém dně formuláře je tlačítko pro odeslání formuláře. Kliknutím na tlačítko se před uživatelem otevře okno s to-do listem.

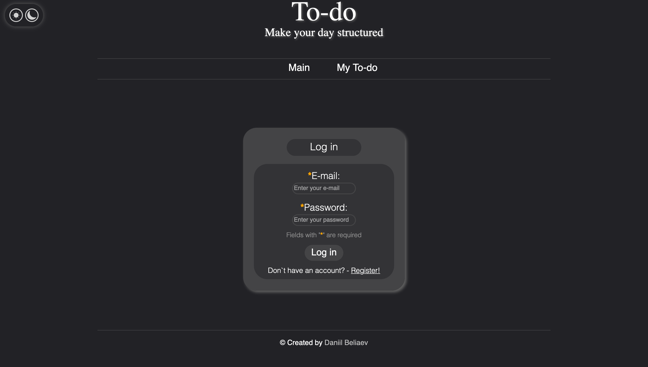
**To-do stránka.** Uprostřed obrazovky je okno to-do listu. V horní části okna je vstupní pole pro přidání úkolu, vpravo je tlačítko pro jeho přidání. Kliknutím na tlačítko se úkol zobrazí níže. Napravo od úkolu jsou tři tlačítka: Odstranit, dokončit, změnit. Po stisknutí tlačítka je úkol spuštěn do spodní části seznamu úkolů.

1. Uživatelská příručka
   1. Popis funkčnosti webu

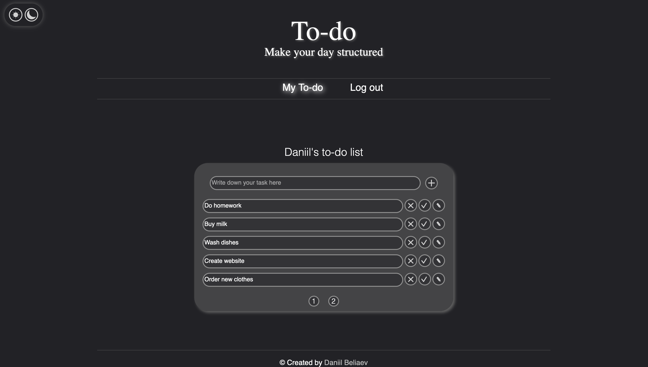
Na hlavní stránce se uživatel setká s uvítacím textem, tlačítkem pro vytvoření nového účtu, navigačním podoknem a tlačítkem skinovatelnost. Pro registraci musíte stisknout tlačítko „Create an account“



Po kliknutí na tlačítko" Vytvořit účet " se otevře registrační okno. Jsou uvedena povinná vstupní pole. Existuje také odkaz na autorizační okno, pokud je uživatel již zaregistrován. Chcete-li se zaregistrovat, musíte vyplnit povinná pole a stisknout tlačítko “Register”.

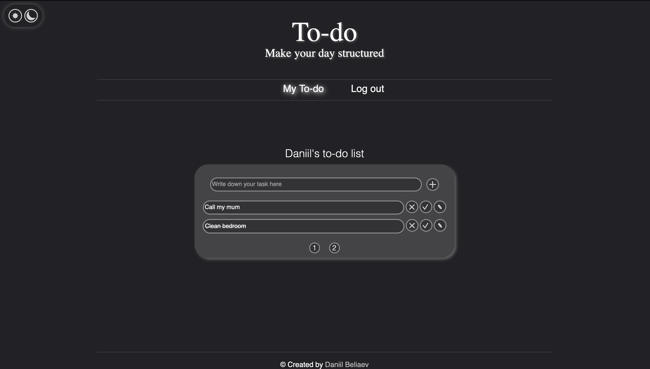


Po dokončení registrace se musí uživatel přihlásit. Chcete-li autorizovat, musíte vyplnit povinná pole a stisknout tlačítko “Log in”.



Po autorizaci je uživateli k dispozici "To-do list".

Uživatel může přidávat, mazat, označovat dokončené a upravovat své úkoly (tlačítka vedle ukolů). Chcete-li přidat úkol, musíte zadat text úlohy do horního vstupního pole a stisknout tlačítko. Pokud se úkoly v seznamu stanou více než 5, budou nové úkoly přeneseny na novou stránku.

Uživatel se také může zrušit autorizaci kliknutím tlačítka “Log out”.

* 1. Ukázky UI webu

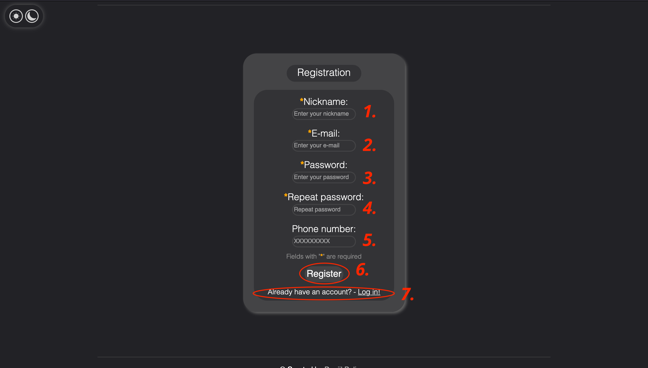


1. Uvítací text

2. Tlačítko pro vytvoření účtu

3. Navigační panel

4. Přepínání světlého a tmavého motivu



1. Vstupní pole pro nickname

2. Vstupní pole pro e-mail

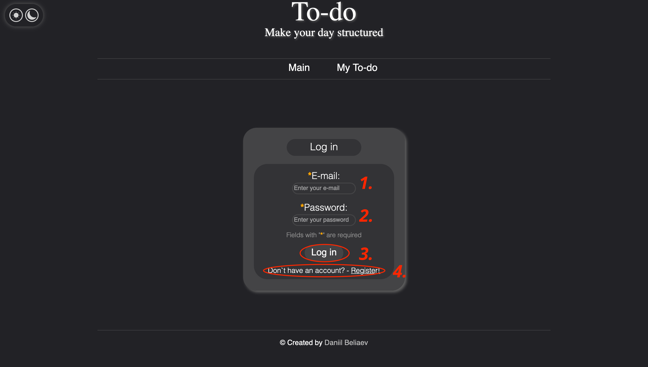
3. Vstupní pole pro heslo

4. Vstupní pole pro opakování hesla

5. Vstupní pole pro mobilní čislo

6. Tlačítko Registrace

7. Odkaz na autorizační okno, pokud je uživatel již zaregistrován

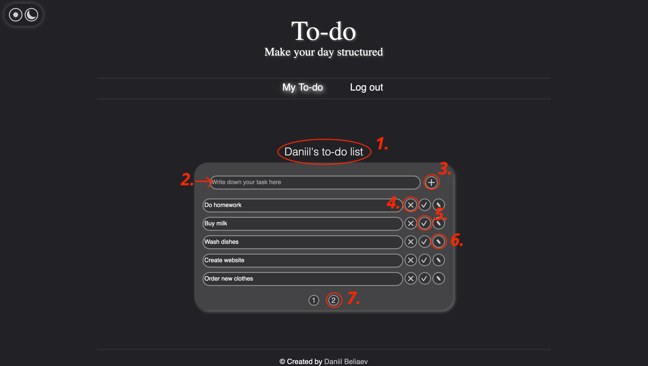


1. Vstupní pole pro e-mail

2. Vstupní pole pro heslo

3. Tlačítko autorizace

4. Odkaz na registrační okno, pokud je uživatel ne zaregistrován



1. Uživatelské jméno

2. Vstupní pole pro zápis úkolu

3. Tlačítko pro přidání úkolu

4. Tlačítko pro odstranění úkolu

5. Tlačítko označující úkol splněný

6. Tlačítko pro úpravu úkolu

7. Tlačítko přepínání stránek

1. Popis implementace
   1. Popis hlavních rysů programu (funkčnost hlavních skriptů)

**Popis architektury.**

Frontend je psán pomocí HTML, CSS, JS. Žádné frameworky nebyli použity. Backend je napsán pomocí PHP a MySQL. Hlavní 4 stránky webu (index.php, registration.php, login.php, todo.php) jsou psány v HTML a CSS a místa, kde je to nutné, zahrnují prvky kódu JS pomocí značky <script> </script> a prvky kódu PHP pomocí značek <?php ?> a <?= ?>.

Kořenová složka "www" ukládá 4 hlavní dokumenty stránek (index.php, registration.php, login.php, todo.php), dále přichází složka "assets" obsahující všechny ostatní potřebné dokumenty. Složka " assets "ukládá složky" css","js","img", " lib."

Složka " css " obsahuje dokumenty:

"style.css " je základní dokument stylů.

"style\_print.css " - dokument stylů pro tisk.

"theme-dark.css " - dokument stylů temného motivu.

"theme-light.css " - dokument stylů světlého motivu.

Složka "img" obsahuje dokumenty:

"del.png " - ikona tlačítka pro odstranění úkolu.

"done.png " - ikona tlačítka úkolu.

"moon.png " - ikona tlačítka Přepnout motiv na tmavý.

"plus.png " - ikona tlačítka Přidat úkol.

"sun.png " - ikona tlačítka pro přepnutí motivu na světlo.

Složka "js" obsahuje dokument:

"validate.js " - dokument validující formuláře.

“DynValidate.js”- validace na všech elementech ve formuláři, při zadávání dat na strance “Registration”

“DynValidateLogin.js” - validace na všech elementech ve formuláři, při zadávání dat na strance “Login”

Složka "lib" obsahuje dokument:

"addTask.php " - dokument, který přidává úkol do databáze

"changeTheme.php " je dokument, který přepíná Barevná témata.

"dark\_theme.php " - dokument měnící hodnotu cookie $\_COOKIE['theme'] = 'dark'.

"light\_theme.php " - dokument měnící hodnotu cookie $\_COOKIE['theme'] = 'light'.

"connect.php " - dokument připojující databázi k webu.

"csrf.php " - dokument ukládání csrf Token

"logout.php " - dokument provádějící autorizaci uživatele.

"paging.php " - dokument implementující stránkování na webu

"reg.php " - dokument registrující uživatele, přidává uživatelská data do databáze.

"saveUpdate.php " je dokument, který ukládá změny úkolu.

"signin.php " – dokument, který delá “log in” uživatele.

"update.php "- dokument zpracovávající tlačítko 'update'

"validate.php " - dokument validující formuláře

"delete.php " - dokument, který odstraní úkol

"done.php " - dokument označující úkol splněný

**Obsluha formulářů.**

Validace je implementována pomocí HTML, JS, PHP.

HTML kontroluje zadaná data prostřednictvím “pattern” a “required”. Validace JS kontroluje každé pole formuláře na minimální délku zadaných znaků, přítomnost znaménka " @ " ve vstupním poli e-mailu. Také se kontroluje, zda byla zadána stejná hesla. V případě nesprávného zadání jakéhokoli pole se zobrazí chyby označující chybu vstupu.

Validace PHP na straně serveru je implementována stejným způsobem. Kontroluje vstupní pole a pokud byla všechna pole zadána úspěšně, jsou data formuláře odeslána na server.

**Bezpečnost.**

Implementována ochrana proti útokům XSS a CSRF. Pomocí funkce PHP htmlspecialchars() je dezinfikováno každé vstupní pole. Pomocí CSRF tokenu je implementována ochrana uživatelských dat při přidávání a úpravách úkolů v seznamu úkolů.

**Algoritmy zpracování dat.**

1. Registrace

Registrace je implementována ve dvou dokumentech.

"registration.php " - registrační formulář a výstup chyb. Po zadání všech uživatelských dat a absolvování tří typů validace, uživatelská data jsou zpracovávány PHP skriptem a odeslány na server (dokument “reg.php”). Pokud některá ze zadaných dat není zadána správně, všechna pole zůstanou vyplněna kromě polí pro zadání hesla. Nejprve je kontrola, zda byl formulář odeslán. Pokud ano, pak informace z každého pole prochází htmlspecialchars(). Dále prostřednictvím přítomnosti e-mailu v databázi je kontrola, pokud je uživatel již zaregistrován. Pokud ano, zobrazí se zpráva přes $\_SESSION ['message'] = 'Registration has not been completed'. $\_SESSION['message\_email'] = 'User with this email already exist'. Pokud ne, pak se uživatelská data spolu s heslem hashovaným přes password\_hash() pomocí dotazu MySQL přidají do tabulky "users" databáze e-mailů. Tabulka obsahuje pole "id", "nickname", "email", "hash", "téma". Po úspěšné registraci uživatel uvidí zprávu ' Registration was successful!'uložené v $\_SESSION ['message\_success'] a bude přesměrován na stránku “login.php”.

1. Autorizace

Autorizace je taky implementována ve dvou dokumentech.

"login.php " – autorizační formulář a výstup chyb. Pokud některá ze zadaných dat není zadána správně, všechna pole zůstanou vyplněna kromě polí pro zadání hesla. Nejprve je kontrola, zda byl formulář odeslán. Pokud ano, pak informace z každého pole prochází htmlspecialchars(). Dále pomocí dotazu mysql zkontroluje, zda existuje uživatel, jehož data byla zadána. Pokud ne, uživatel uvidí zprávu 'User with this e-mail does not exist' uloženou v $\_SESSION ['message\_email\_login']. Dále je kontrola shody zadaného hesla implementovaná prostřednictvím password\_verify (). Pokud se hesla neshodují, uživatel uvidí zprávu 'Password was written wrong' uloženou v $\_SESSION ['message\_password\_login'].

Pokud byl uživatel nalezen v databázi a hesla se shodovala, pak se do $\_SESSION['user'] zapíše id, nickname, email uživatele. Po úspěšné autorizaci, uživatel je přesměrován na stránku " todo.php"

1. Log out

Po kliknutí na odkaz "Log out" na stránce " todo.php "spustí skript v dokumentu" logout.php". V něm je uživatel odstraněn z session pomocí unset ($\_SESSION['user']). Další je přesměrování zpět na dokument " todo.php".

1. Přidávání úkolů

Po zadání úkolu a kliknutí na tlačítko " + "na stránce" todo.php" spustí skript v dokumentu "addTask.php". Dokument prochází dezinfekci zadaného textu úkola pomocí htmlspecialchars(). Dále je zkontrolováno, zda byl do vstupního pole zadán nějaký text. Pokud ne, uživatel uvidí zprávu 'Enter your task' uloženou v $\_SESSION ['message\_task'], skript ukončí svou práci pomocí příkazu exit(). Pokud byl zadán text, pak se pomocí mysql dotazu do tabulky "tasks" databáze přidá id uživatele a samotný úkol. Dále je přesměrování zpět na " todo.php".

1. Odstranění úkolů

Po kliknutí na tlačítko "Odstranit" na stránce " todo.php" spustí skript v dokumentu "delete.php". Adresa url předá id úkolu, který je třeba smazat. Dále je toto id načteno z globálního pole $\_GET. Pomocí dotazu mysql se z tabulky "úkoly" databáze odstraní úkol s požadovaným id. Dále je přesměrování zpět na " todo.php".

1. Označení úkolů dokončenými

Po kliknutí na tlačítko "Hotovo" na stránce " todo.php "spustí skript v dokumentu" hotovo.php". Adresa url předává ID úkolu a stav, který je třeba označit jako dokončený. Stav v tomto případě je číselná hodnota uvedená v tabulce "úkoly". Pokud je stav 1, pak úkol není označen jako dokončený, pokud je 0, je označen jako dokončený (přeškrtnutý). Probíhá kontrola aktuálního stavu úkolu a pokud je stav 0, pak se jeho hodnota změní na 1 a naopak. V závislosti na hodnotě stavu úlohy v dokumentu "todo.php " zobrazuje úkol buď přeškrtnutý nebo ne. Dále je přesměrování zpět na " todo.php".

1. Úprava úkolů

Po kliknutí na tlačítko "update" na stránce "todo.php" spustí skript v dokumentu "update.php". Adresa url předává id úkolu a stav "updateStatus", který je třeba změnit. Stav "updateStatus" je v tomto případě číselná hodnota uvedená v tabulce "tasks". Pokud "updateStatus " je 0, pak se úkol neaktualizuje, pokud je 1, pak úkol vstoupí do režimu aktualizace. Skript v dokumentu " update.php " změní tento stav pomocí mysql dotazu. Pokud je úkol v režimu aktualizace, pak na stránce " todo.php " se zobrazí místo úkolu input s textem úkolu, který lze změnit a tlačítko "Uložit". Po kliknutí na tlačítko "uložit", skript se spustí v dokumentu "saveUpdate.php". V něm upravený text prochází kontrolou přes htmlspecialchars(). Dále je kontrola, zda byl vůbec zadán nějaký text, pokud ne, pak uživatel uvidí zprávu "Enter your task", dojde k přesměrování zpět na " todo.php", práce skriptu je přerušena příkazem exit(). Pokud byl zadán text, pak pomocí dotazu mysql v tabulce "tasks" databáze použije pole "task" novou hodnotu. Dále "updateStatus" zpět bere hodnotu 1 a uživatel je předán zpět na " todo.php".

1. Skinovatelnost

Na každé stránce webu vlevo nahoře jsou dvě tlačítka s obrázkem slunce a měsíce. Pokud uživatel klikne na tlačítko "měsíc", spustí se skript v dokumentu "dark\_theme".php". Odstraní starou hodnotu $\_COOKIE['theme']="light" a nastaví novou hodnotu $\_COOKIE['theme']="dark". K tomu dochází pomocí funkce setcookie(). Další je přesměrování uživatele na stránku, na které byl umístěn. Skript funguje na stejném principu v dokumentu "light\_theme.php", který mění hodnotu cookies z "dark" na "light". Na každé stránce webu je vložen kód dokumentu "changeTheme.php" pomocí příkazu require. V tomto dokumentu skript zkontroluje, zda je soubor cookie vůbec nastaven, a pokud ne, nastaví výchozí hodnotu na "light". Dále je kontrola, která hodnota souboru cookie je nastavena, a pokud je hodnota "light", pak proměnná $style vezme cestu k dokumentu stylu"theme-light.css" a naopak. Proměnná $style se zasekne na každé stránce ve značce <link> v atributu "href".