Съдържание

[1. Увод 1](#_Toc191916792)

[2. Изложение 1](#_Toc191916793)

[3. Заключение 1](#_Toc191916794)

[4. Литература: 2](#_Toc191916795)

# увод

# Цели на дипломния проект

Целта на дипломния проект е да разгледа и разработи система за удостоверяване на самоличността и оторизация на потребителите в уеб приложения. В условия, в които интернет заплахите продължават да нарастват, надеждното удостоверяване на потребителите става изключително важно за защитата на личните данни и информация. Проектът цели да съчетае теоретични познания с практическа разработка, осигурявайки стабилна основа за съвременни уеб приложения, които да предпазват потребителската информация.

**Основни цели:**

1. **Изследване на съществуващите решения:**
   * Проектът ще изследва различни методи за удостоверяване на самоличността, като основни технологии и подходи включват OAuth, OpenID Connect и JWT (JSON Web Tokens). Важно е да се анализират предимствата и недостатъците на всяка от тези технологии, за да се прецени коя е най-подходяща за реализиране на системата.
2. **Създаване на система за регистрация и вход:**
   * Практическата част на проекта ще се състои в разработването на система за регистрация и вход на потребители в уеб приложение, реализирано на базата на Django. Системата ще включва функции за управление на акаунти, затова ще бъде проектирана да поддържа различни роли и права на достъп, което ще позволи на администраторите да контролират достъпа до ресурси.
3. **Осигуряване на сигурност на данните:**
   * Важна цел на дипломния проект е да се приложат съвременни практики за сигурност, включително хеширане на пароли и механизми за защита от атаки, като SQL инжекции и XSS (Cross-Site Scripting). Това ще гарантира безопасността на данните, като същевременно осигурява удобство на потребителите при взаимодействие с приложението.
4. **Разработка на интуитивен потребителски интерфейс:**
   * Също така, проектът включва създаването на потребителски интерфейс, който да е интуитивен и адаптивен, осигурявайки оптимално изживяване на потребителите независимо от устройството, което използват (десктоп, таблет или мобилен телефон). Интерфейсът ще бъде проектиран с акцент върху навигация и визуална яснота.
5. **Оценка на възможностите за интерактивност:**
   * Проектът ще разгледа как да осигури динамично генерирана информация за потребителите, в зависимост от техния статус и роля в системата. Това включва показване на персонализирано съдържание след успешна автентикация, което ще увеличи ангажираността на потребителите и ще подобри потребителското изживяване.

# Теоретичен фон

## 1. Удостоверяване на самоличността и оторизация

Удостоверяването на самоличността и оторизацията са два ключови аспекта в сигурността на уеб приложенията. Те обхващат процесите на проверка на самоличността на потребителя и определяне на правата му за достъп до ресурси.

* **Удостоверяване на самоличността** се отнася до потвърждаването на идентичността на потребителя. Концепцията включва различни методи, сред които е традиционното удостоверяване чрез потребителско име и парола, но през последните години все повече се използват технологии, свързани с многофакторна автентикация. Този процес е критичен, тъй като правилното удостоверяване предотвратява несанкциониран достъп до чувствителна информация.
* **Оторизацията** е етапът след удостоверяването, при който системата решава кои ресурси и операции са достъпни за конкретния потребител. Това обикновено включва задаване на роли и права, като например администратор, редактор, и обикновен потребител. Успешната комбинация от тези две функции осигурява безопасността на данните и минимизира рисковете от злоупотреби.

## 2. Методи за удостоверяване на самоличността

С развитието на интернет технологиите, методите за удостоверяване са се еволюирали и могат да се класифицират в основни категории:

1. **Удостоверяване с потребителско име и парола:** Това е най-често използвания метод, и се основава на нещо, което потребителят знае. Главният недостатък е, че паролите могат да бъдат отгатнати или откраднати.
2. **Многофакторна автентикация (MFA):** Този метод комбинира два или повече различни типа удостоверяване, например парола и код, изпратен на мобилен телефон. MFA значително увеличава сигурността, като минимизира риска от компрометиране на данни.
3. **Биометрично удостоверяване:** Използва уникални физически характеристики на потребителя, като пръстов отпечатък, лице или ирис. Тази технология е устойчива на фалшификация, но изисква специализирана технология за събиране и обработка на данни.
4. **Токени и сертификати:** Включва генерация на токени, които се използват за удостоверяване. Например, JSON Web Tokens (JWT) дават възможност за безопасно предаване на информация за удостоверяване и права за достъп.
5. Интеграция с социални мрежи
6. Добавянето на опция за вход чрез социални мрежи (например Google, Facebook) би улеснило потребителите при регистрация и вход, тъй като те не биха трябвало да запомнят допълнителни пароли. Това предлага и допълнителен слой на сигурност, ако данните на социалните профили са защитени.
7. Регистрация, възстановяване и управление на профили

Услугите за регистрация и управление на профили предоставят:

Процес на лесна и бърза регистрация, който включва механизми за потвърдителни имейли и валидиране на потребителски данни.

Възможност за възстановяване на пароли с безопасни методи, включително процедура на повторно задаване, прилагана след потвърждаване на имейл адреса.

## Оторизация и управление на права

Оторизацията в контекста на уеб приложения се реализира чрез модели за управление на права (RBAC - Role-Based Access Control). В този модел:

* Потребителите получават определени роли, и всяка роля има свои права за достъп до ресурси.
* Тази система опростява управлението на права и позволява лесна интеграция на нови потребители.

Двете основни концепции в управлението на права са:

* **Права на ресурси:** Определят какви действия могат да извършват потребителите с определени ресурси, като четене, писане или изтриване.
* **Политики за достъп:** Системи, определящи как и кога потребителите могат да получат достъп до данни или услуги преди техния идентификатор.

**Актуални технологии за удостоверяване и оторизация**

С развитието на уеб технологиите, нови решения съществуват, за да отговорят на растящите нужди за сигурност и удобство. Някои от най-често използваните технологии и протоколи включват:

* **OAuth 2.0:** За разрешаване на достъпа и позволява на едно приложение да получава достъп до потребителски данни от друго приложение, без да разкрива пароли.
* **OpenID Connect:** Разширение на OAuth 2.0, което добавя слой на удостоверяване. Технологията позволява на потребителите да влизат в приложения с помощта на съществуващи идентичности от популярни платформи като Google и Facebook.
* **SAML (Security Assertion Markup Language):** Предоставя способи за Single Sign-On (SSO), което позволява на потребителите да се удостоверяват веднъж и да получават достъп до множество приложения.

# Технологичен анализ

В рамките на технологичния анализ на дипломния проект „Удостоверяване на самоличността и оторизация в уеб приложения“, е необходимо да се разгледат основните технологии, платформи и инструменти, които ще се използват за реализиране на практическите и теоретични аспекти на проекта. Технологиите, избрани за реализиране на системата, трябва да съответстват на актуалните стандарти за практики в областта на уеб разработката и да предлагат надеждни механизми за сигурност.

Аргументация за избора:

Бърза разработка: Django следва принципите на "не повтаряй се" (DRY), което намалява времето за разработка и улеснява поддръжката на кода.

Сигурност: Фреймуъркът предлага вградени защити срещу често срещани уязвимости, като SQL инжекции, Cross-Site Scripting (XSS) и Cross-Site Request Forgery (CSRF).

Модуларност: Django разполага с приложения, което позволява лесно добавяне на нови функционалности, което е особено полезно за бъдещо разширение на системата.

1. Основни технологии

Django Framework:

Django е един от най-популярните уеб фреймуърци за разработка на приложения на Python. Той предлага модулна структура, която позволява бързо разработване на приложения с вградени механизми за сигурност. Основни предимства на Django включват:

Сигурност: Django предлага множество вградени инструменти за защита срещу често срещаните уеб заплахи, включително XSS, CSRF, SQL инжекции и др.

Технологичен стек: Django е вграден с множество библиотеки и модулни компоненти, което улеснява интеграцията на функционалности.

Административен интерфейс: Django предоставя готов административен интерфейс, който може да се използва за управление на потребители и данни.

SQLite/MySQL/PostgreSQL:

За управление на базата данни в проекта може да се използва вградена SQLite база или по-производителни решения като MySQL или PostgreSQL. Изборът на конкретна база данни зависи от критериите на приложението:

SQLite: Подходяща за малки приложения и разработка, тъй като не изисква инсталиране на отделен сървър. Позволява бързо стартиране и тестове.

MySQL/PostgreSQL: Препоръчва се за по-мащабни приложения, когато е необходима сложна работа с данни, транзакции и разширени функции.

2. Технологии за удостоверяване

При разработването на системата за удостоверяване на самоличността и оторизацията, е важно да се използват утвърдени протоколи за сигурност и автентикация:

OAuth 2.0:

OAuth 2.0 е протокол за авторизация, който позволява на приложение да публикува и получава данни от друг ресурс, без да разкрива паролата на потребителя. Важни аспекти на OAuth 2.0:

Поддържа различни типове клиентски приложения (мобилни, уеб и клиентски приложения).

Осигурява деликатно управление на познанията за потребителска информация.

Оптимизира интеграцията с различни уеб услуги и API.

JSON Web Tokens (JWT):

JWT предлага безопасно представяне на информация, която се предава между сървъра и клиента. JWT е компактен и идеален за работа с уеб приложения поради следните характеристики:

Възможност за предаване на информация в URL-параметри или заглавия на HTTP.

Сигурност чрез цифрово подписване, което предотвратява множество видове атаки.

Лесна работа с клиентски приложения, позволявайки безсесийно удостоверяване.

3. Потребителски интерфейс и опит

Системата за удостоверяване ще включва потребителски интерфейс (UI), който трябва да бъде интуитивен и адаптивен. За целта могат да се използват различни технологии и библиотеки:

Регистрация на потребител:

При регистрация, потребителят попълва необходимите данни (потребителско име, парола и др.).

Параметрите следва да бъдат валидирани (например, паролата трябва да отговаря на определени правила за сложност).

Паролата на потребителя се хешира с подходящ алгоритъм (например, PBKDF2 или bcrypt) преди да бъде запазена в базата данни, за да се осигури допълнителна защита.

HTML/CSS:

Основни технологии за изграждане на структурирани и стилни уеб интерфейси. Чрез използването на CSS фреймуърци, като Bootstrap, можем да осигурим адаптивен дизайн, който работи добре на различни устройства.

JavaScript и AJAX:

За интерактивност на приложението и динамично зареждане на данни ще се използват JavaScript и AJAX. Това ще позволи на интерфейса да комуникира с сървъра, без да се презарежда страницата, което подобрява

# Приноси на дипломния проект

1. Въведение

Приносите на дипломния проект "Удостоверяване на самоличността и оторизация в уеб приложения" са от съществено значение не само за саморазвитието на ученика, но и за разработването на устойчиво решение, адресиращо съвременни предизвикателства в областта на киберсигурността и уеб технологии. В следващите раздели ще разгледаме основните приноси на проекта, структурата на разработената система, както и значението на постигнатите резултати.

2. Основни приноси

Създаване на работеща система за удостоверяване:

Проектът осигурява работеща система за удостоверяване на самоличността и оторизация в уеб приложения, реализирана на базата на Django. Тази система предоставя ефективни механизми за регистрация, вход и управление на потребители, което я прави полезна за всяко уеб приложение, изискващо автентикация на потребителски данни.

Изграждане на интуитивен интерфейс:

Чрез разработването на интерфейс, фокусиран върху потребителското изживяване, проектът осигурява на потребителите лесно и удобно взаимодействие със системата. Интуитивната навигация и адаптивният дизайн позволяват на потребителите да се ориентират бързо и ефективно, независимо от устройството, което използват.

Изучаване и прилагане на съвременни технологии:

Проектът елиминира традиционни методи за удостоверяване и внедрява актуални технологии като OAuth 2.0 и JSON Web Tokens (JWT), което значително повишава нивото на сигурност на данните. Учениците не само, че научават за тях, но и прилагат на практика принципите и механизмите, които те предлагат.

Система за управление на роли и права:

Важен принос на проекта е внедряването на механизми за управление на роли и права на достъп до ресурсите. Системата дава възможност на администраторите да контролират кой и какво може да прави в интерактивната среда, осигурявайки по-добра защита на данните.

Обогатяване на знанията в областта на киберсигурността:

Като част от дипломния проект, учениците попълват знанията си за киберсигурността, научавайки как да защитават информационни системи и потребителски данни от нападатели. Това знание е от решаваща важност в съвременната дигитална ера, където заплахите са част от ежедневието.

Документиране на процеса на разработка:

Проектът включва концентриран документален процес, който предоставя ясни указания за стъпките, включени в разработката на системата. Това е полезно не само за бъдещи подобрения на проекта, но и за нови разработчици, които може да искат да пресъздадат или подобрят системата.

3. Практически резултати и приложения

Функционален уеб проект:

Реализираната система демонстрира работата на основните функции, необходими за удостоверяване и оторизация на потребители. Системата е тествана на реални данни, което позволява да се зачитат реални сценарии на работа.

Потенциал за разширение и интеграция:

Проектът предоставя основа за бъдеща интеграция на допълнителни функции, като например услуги за забравена парола, параметрична аутентикация, и двафакторна автентикация, което в значителна степен повишава сигурността на потребителските акаунти.

Образователна стойност:

Разработката на проекта предлага ценен опит в сферата на уеб програмното осигуряване. Учениците, работещи по проекта, развиват своите технически умения и получават практически опит в решаването на комплексни проблеми.

# Заключение

# Литература:

1. Колисниченко, Денис, Адаптивен уеб дизайн с Bootstrap, Асеневци, С 2019
2. D.K Academy, Python – практическо програмиране. Асеневци, С 2022
3. Донълдсън, Тоби. Бързо ръководство:Програмиране с Python.АлексСофт, С 2017
4. Есканази, Аврам. Софтуерни техологии. КЛМН, С 2006
5. https://softuni.bg/blog/what-is-django
6. 2. https://cynoteck.com/bg/blog-post/flask-vs-django/
7. 3. https://bg.savtec.org/articles/coding/the-basics-of-rest-and-restful-api-development.html
8. 4. https://urocibg.eu/
9. 5. https://www.django-rest-framework.org/
10. 6. https://softuni.bg/blog/vue-js-january-2021
11. 7. https://bg.wikipedia.org/wiki/MySQL
12. 8. https://priobshti.se/article/strategii-v-pomosht-na-prepodavaneto/taksonomiya-na-blum-ili-kak-da-napravim-uroka-v
13. 9. https://html.w3schools.bg
14. 10. https://www.inventum.bg/web-design/css/