# ЧАСТНА ПРОФЕСИОНАЛНА ГИМНАЗИЯ ПО ДИГИТАЛНИ НАУКИ

# „СОФТУНИ СВЕТЛИНА“

# 

# 

# 

# **ДИПЛОМНА РАБОТА**

# ЗА ПРИДOБИВАНЕ НА ПРОФЕСИОНАЛНА КВАЛИФИКАЦИЯ

# НА ТЕМА:

“Генератор на пароли”

ИЗГОТВИЛ: Калоян Коцев НАУЧЕН РЪКОВОДИТЕЛ: Николай Сапунов

# 

# 

# 03.05.2023 г.

# ГР. СОФИЯ

# **Съдържание**.............................................................................2Увод

1. Въведение.......................................................................................................3

3. Типове генератори на пароли.......................................................................4

4. Описание на функционалността..................................................................6

5. Използвани технологии................................................................................8

6. Сигурност......................................................................................................9

7. Тестове.........................................................................................................10

Идея за проекта................................................................................................11

8. Ограничение и бъдещо развитие................................................................12

9. Приложения..................................................................................................13

10. Заключение................................................................................................14

11. Ресурси.......................................................................................................15

Въведение

Генераторът на пароли е софтуерен инструмент, който генерира уникални и здрави пароли за потребители. В днешния цифров свят, където всичко - от банкирането до социалните мрежи - се прави онлайн, е от съществено значение да имате силни и сигурни пароли, за да предпазите онлайн акаунтите и личната си информация от неправомерен достъп. Пароли, които са лесни за отгатване или за разбиване, могат да доведат до компрометиране на данните, кражба на самоличност и финансови загуби.

Генераторите на пароли предлагат решение на този проблем, като създават сложни пароли, които са трудни за отгатване или разбиване. Генераторите на пароли могат да създават пароли, които са по-дълги, съдържат комбинация от символи и са напълно случайни, което ги прави по-сигурни от пароли, създадени от самите потребители.

Целта на тази документация е да предостави обзор на инструмент за генериране на пароли, включително неговите функции, функционалност и имплементация. Тази документация е предназначена за всички, които се интересуват от използването или разработването на генератор на пароли, включително спонсори на проекти, разработчици и крайни потребители.

В раздела "Типове генератори на пароли" ще разгледаме различните видове генератори на пароли, включително сложността на генерираните пароли и начините, по които те могат да бъдат използвани. Ще предоставим и сравнителен анализ на различни генератори на пароли, за да ви помогнем да изберете най-подходящия за вашите нужди.

# Типове генератори на пароли

Има много различни видове генератори на пароли.

Най-критичният фактор, който трябва да имате предвид, преди да изберете такъв, е къде ще го използвате. Можете да използвате генератор на пароли на всяко устройство, което има достъп до интернет.

Генераторът на псевдослучайни числа (PRNG) - се отнася до алгоритъм, който използва математически формули за създаване на последователности от произволни числа. PRNG генерират поредица от числа, приближаващи свойствата на произволни числа. PRNG започва от произволно начално състояние, използвайки начално състояние. Много числа се генерират за кратко време и могат да бъдат възпроизведени по-късно, ако е известна началната точка в последователността. Следователно числата са детерминистични и ефективни.

Генератор на естествени случайни числа (TRNG)- известен също като хардуерен генератор на случайни числа (HRNG), не използва компютърен алгоритъм. Вместо това той използва външна непредвидима физическа променлива, като радиоактивно разпадане на изотопи или статична вълна, за да генерира произволни числа.

Криптографски защитени генератори на псевдослучайни числа(CSPRNG) - те имат изискванията на CSPRNG попадат в две групи: първо, че преминават тестове за статистическа случайност; и второ, че издържат добре при сериозна атака, дори когато част от тяхното първоначално или работещо състояние стане достъпно за нападател. Случайните последователности, генерирани от всеки CSPRNG, се идентифицират уникално чрез следните три характеристики: (a) висока ентропия; (b) без повторение в генерираните низове; в) нулева корелация.

Grammatical Evolution (GE) е био-вдъхновен инструмент за машинно обучение (ML), базиран на населението, който използва граматиките на Backus-Naur Form (BNF) за генериране на правни структури за различни проблемни области. В тази изследователска работа ние изследваме GE като потенциален източник на ентропия за генериране на първоначално семе с висока ентропия, тъй като всеки еволюционен цикъл е в състояние да произведе различно семе. Ние предлагаме базиран на GE PRNG и базиран на GE CSPRNG, като и двата могат да генерират произволни последователности, които се придържат към покриване на всички стандарти за отлична псевдослучайност.

# Описание на функционалността

Всички генератори имат едно общо помежду си, те работят, като използват това, което ни е известно като семена.

Генераторите на пароли работят, като използват това, което е известно като семена.

Семената се използват за генериране на случаен низ от знаци, които се използват за създаване на вашата парола.

След като паролата бъде генерирана, семената могат да бъдат изхвърлени, така че да не се налага да я помните.

Начинът, по който се използва семената, ще се различава в зависимост от избрания от вас генератор на пароли.

Някои помощни програми ще използват произволно семе, което не може да бъде предвидено. Други са проектирани с предварително дефинирани алгоритми, които могат да бъдат проектирани обратно, ако някой има достъп до семето, което е използвал за създаване на своите пароли.

Семената се използват за генериране на случаен низ от знаци, които се използват за създаване на вашата парола.

След като паролата бъде генерирана, семената могат да бъдат изхвърлени, така че да не се налага да я помните. Начинът, по който се използва семената, ще се различава в зависимост от избрания от вас генератор на пароли.

Някои помощни програми ще използват произволно семе, което не може да бъде предвидено. Други са проектирани с предварително дефинирани алгоритми, които могат да бъдат проектирани обратно, ако някой има достъп до семето, което е използвал за създаване на своите пароли.

Този проект работи на принципа на генератор на псевдослучайни числа. Функцията (Math.Random), която се използва в проекта за “случайното” генериране на символите , работи с математични формули, за да имитира случайност.

Тези математически формули могат да работят само, ако бъдат зададени периметрите на паролата(букви,цифри,знаци и тн.), за това този тип генератори биват наречени “псевдо генератори”

Проектът работи като потребителя задава дали ,да има малки и големи букви, да има ли специални знаци, цифри и колко ще е дълга паролата ,за да може тя да работи.

Във следващата част ще бъдат обяснени технологиите ползвани проекта и тяхното предназначение.

Използвани Технологии 1.Javascript В този проект Javascript бива използван, за да може генератора да може да изпълнява своите задачи. Интерактивната част на този проект работи като създадем функции, които работят с “аsсii” таблицата, катo чрез функциите “Math.random” и “Math.floor”. Начинът по който генератора създава паролата като първо проверява броя на символите ,после проверява какъв вид символи потребителя задава преди да почне да генерира паролата. Паролата се генерира ,чрез вложен цикъл който оставя само нужните случайни символи. 2.HTML HTML бива използван в този проект, като структура по която да бъдат написани Javascript-a и CSS-a. В него биват създадени: облика на проекта и бутоните които биват натискани ,за да работи. 3.CSS CSS е използван за стила и цвета на проекта. Чрез него също така влиза и библиотеката ”Аnimate” която е използвана за шрифта и снимката на бутона.

# 

# Сигурност

Не всички „генератори на случайни пароли“ създават наистина произволни резултати. Ако използвам генеричен генератор на произволни пароли, за да създам 10 000 нови пароли, генерираните от мен пароли са резултат от феномен, наречен псевдослучайни, което означава, че резултатите изглеждат произволни, когато всъщност не са. Стив Уорд, професор по компютърни науки и инженерство в лабораторията по компютърни науки и изкуствен интелект на Масачузетския технологичен институт, казва, че традиционните компютърни системи не са добри при генерирането на произволни резултати. „Те са детерминистични, което означава, че ако зададете един и същи въпрос, ще получите един и същ отговор всеки път“, казва той. „Всъщност такива машини са специално и внимателно програмирани да елиминират произволността в резултатите. Те правят това, като следват правила и разчитат на алгоритми, когато изчисляват. Според Уорд една напълно детерминистична машина не може да генерира наистина произволни числови последователности, защото следва същия алгоритъм, за да произведе своите резултати. „Обикновено това означава, че започва с общ „начален“ номер и след това следва модел“, казва той. Следователно вашите рандомизирани пароли не са наистина произволни, защото са създадени от същия алгоритъм. За да може компютърът ми да генерира наистина произволен резултат от хвърляне на зарове, например, компютърът ми трябва действително да хвърли самия зар. Тъй като е изключително трудно да се свърже истински зар с компютър (и очевидният факт, че компютрите нямат ръце), за компютрите е по-лесно да генерират наистина произволни резултати, ако разчитат на непредвидими процеси, като физически феномени: движенията на вашия компютърна мишка, фонов или атмосферен шум, радиоактивен разпад или моментни снимки на лава лампи.

Тестове За тестването този проект бяха изпозлвани методите: debug и unit testing. Debug Той беше много използван по време на работата по този проект, понеже се стигаше до грешки ,които не можеше да бъдат намерени и се наложи да бъде debug-нато Помогна в изчистване то на много бъгове от проекта. Unit testing Това беше метода на тестването ,който предимно беше използван по време на писане на дипломната работа. Помогна много, защото в програмата има части от нея ,които са по-лесни за тестване едно ,по едно от колкото да бъде проверявана цялата.

Идея на проекта

Проекта е създаден с идеята да даде на хората спокойствие от кибератаките ,които се случват доста често в днешно време . Много хора отделят купища време и пари ,за да са сигурни, че интернет съществуването им да може да мине с възможно най-малко проблеми. Поради тази причина аз реших да създам този генератор на пароли, за да може да се чувствам по спокоен в интернет, без да мисля толкова много за какво може да ми се случи ,ако не съм твърде внимателен.

Ограничения и бъдещо развитие

Нищо в програмирането не е ограничено. Един програмист винаги има силата да направи каквото си поиска от нищото. Вярвам, че този проект има много да се развива. Може да се направи да е с криптирани пароли ,може по-хубав интерфейс да му бъде направено и тн.

Приложения

Генераторите на пароли са ценни по няколко причини.

Те ни позволяват да създаваме дълги, сложни пароли. Генераторите на пароли също са от полза, ако имате проблеми с измислянето на пароли, които отговарят на специфични изисквания (напр. поне 12 знака, цифри, символи и т.н.).

Те ви позволяват да създавате различни пароли за различни уебсайтове, така че ако един бъде компрометиран, останалите да останат в безопасност. Това може да помогне за предотвратяване на хакерите да пробият една система и да получат достъп до всичките ви акаунти.

Най-важното е, че генераторите на пароли са широко достъпни в различни компютърни платформи. Те могат да бъдат достъпни от уебсайтове, настолни компютри, лаптопи, таблети, смартфони и други мобилни устройства.

Това е особено важно, ако често влизате в интернет от обществени компютри. Когато случаят е такъв, често сте принудени да създавате слаби пароли или рискувате системният администратор да излезе от системата.

Kaк Работи

Кода работи като от Html-a инициализира полетата и бутоните, javascript-а инициализира промените и върши работата на проекта а CSS-a е за дизайна на проекта. Във HTML имаме въведени 6 бутона и едно поле където изписва резултата от получената парола. 4 от бутоните са тикчета и не вършат нищо освен да казват на програмата да не слага тaкива символи в паролата. Другите два бутона са бутоните Generate Password и бутона за копиране. В javascript-a може да видим интерактивната част от проeкта. Вътре в него има инициализирани DOM променливи, че да може да се достъпва с html-a .Към тях има функции, които помагат за избирането на знак чрез ”ascii” таблицата, която им помага да извадят съответният знак. С константата randomFunc биват инициализирани всички функции. Във конструкцията на кода също така има два EventListener-a , които ни помагат с натискането на бутоните. Единият е за копиране на получената от потребителя парола. В същия EventListener също така излиза и нотификация при копиране на паролата. Другият е за генерирането на паролата. При генерирането на паролата трябва да бъде направена проверка . Проверка през която трябва да се премине, е дали типът данни и дължината, които един потребител може да изисква. След като се мине през тази проверка приложението минава през вложен цикъл който пълни символите в масив. Тези символи след това биват slice-нати от 0 до дължината на паролата. И накрая тя бива изпринтирана. В CSS-a бива взета библиотеката”animate” за шрифта и за снимката на бутона за копиране. Там също може да видим и цветовете използвани за проекта. Чрез CSS-a проекта получава своя лилав цвят.

Заключение

Паролите са жизнено важна част от нашето съществуване в 21 век, ако не държим ние на тях, няма кой друг. Заради това аз реших да направя този генератор на псевдослучайни пароли, че да се чувстваме по-спокойно докато бърникаме из интернет. Ако един човек отделяше толкова време на паролите си, толкова колкото на телефона си, кибер атаките най-вероятно щяха да са намалели драстично ,до сега.

Препратки <https://www.passcamp.com/blog/the-complete-guide-to-password-generators-what-they-are-and-why-you-need-one/> <https://www.dashlane.com/blog/how-random-password-generators-work> https://www.knowledge-share.eu/en/patent/quantum-random-number-generator/?gclid=Cj0KCQjwr82iBhCuARIsAO0EAZyv-B3zX8HiI23Y-1Fw5W8RcqUPwJy3Zjw14kFHAmIjfvvyZ5Io2iIaAnUpEALw\_wcB <https://www.nature.com/articles/s41598-022-11613-x> https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\_ Objects/Math/random https://www.geeksforgeeks.org/pseudo-random-number-generator-prng/ <https://codemyui.com/> https://stackoverflow.com/