

8.2. Файл index.php

Введение

Данный файл является **Индексным файлом** — это файл главной страницы директории на сайте, то есть тот файл, который загружается, когда посетитель обращается напрямую к директории. В нашем приложении Файл index.php - это точка входа.

Логика index_test.php

- **ЕСЛИ** ядро приложение не установлено (т.е. отсутствуют файлы **version.txt** и **config.ini** в директории **/includes**), то приложение выведет в окне **/library/install** к файлу **install.php**
- **ИНАЧЕ** проверит переменную **\$default_site** в файле **config.ini**
- **ЕСЛИ** переменная **\$default_site = пустое значение**, то приложение выведет в окне индексный файл, первой из установленных конфигураций
- **ИНАЧЕ** получив значение переменной **\$default_site** из файла **default_site** перенаправит к **install.php** - индексный файл установленной конфигурации, указанный пользователем.

Файл index_test.php

Создадим Файл **index_test.php** в **/ucms** с кодом:

[index_test.php](#)

```
<!-- test/index_test.php -->

<?php
/* подключим файл с константами, для определения корневой директории приложения
*/
require_once __DIR__ . "../includes/cons.php";

/* Присваиваем переменной $file_config значение константы FOLDER
(а это директория ucms, записанная в файле cons.php) и добавляем
'/includes/config.ini' - директорию includes и искомый файл
config.ini */
$file_config = FOLDER . '/includes/config.ini';
/* Если config.config.ini существует */
if (file_exists($file_config)) {
    /* выводим запись о существовании */
    echo "Файл $file_config существует". "<br/>";
    /* Иначе выводим запись о существовании */
} else {
    echo "Файл $file_config не существует". "<br/>";
}
```

```
}

// Создание и вызов параметров из .ini файла
////////////////////////////////////
/* задаем переменной значение $config
 * parse_ini_file - обрабатывает конфигурационный файл
 * ../../includes/config.ini - путь к config.ini
 * ,true); - обрабатываем с секциями (зпголовками)
 */
$config_default_site = parse_ini_file(FOLDER .
"/includes/config.ini",true);
// Проверьте, успешно ли прочитан ini-файл.
// переменная $config не равно ЛОЖЬ
if ($config_default_site !== false) {
    // Доступ к индивидуальным настройкам

    $default_site =
$config_default_site['default_site']['default_site'];
    echo "Конфигурация по умолчанию: " . $default_site . "<br/>";
} else {
    echo "Ошибка чтения INI-файла.";
}
//exit;
/* зададим переменным файлы и директории */
$file_config = FOLDER . '/includes/config.ini';
$file_version = FOLDER . '/version.txt';

$bcw_site = FOLDER . '/bcw';
$log_site = FOLDER . '/blog';
$video_site = FOLDER . '/video';
$sns_site = FOLDER . '/sns';

/* Если переменные $file_config и file_exists существуют,
то проверим определение переменной конфигурации $default_site
по умолчанию в файле config.ini */
if (file_exists($file_config) && file_exists($file_version)) {
    /* Если переменная $default_site равно "пустое место" */
    if ($default_site == ""){
        /* Проверим, что конфигурация установлена в директории /bcw */
        if (file_exists($bcw_site)) {
            /* Если директория /bcw существует, перенаправим туда и конец скрипта*/
            header("Location: ../bcw/");
            exit;
        }
        /* Проверим, что конфигурация установлена в директории /blog */
        elseif (file_exists($log_site)) {
            /* Если директория /blog существует, перенаправим туда и конец скрипта*/
            header("Location: ../blog/");
            exit;
        }
    }
}
```

```
/* Проверим, что конфигурация установлена в директории /video */
elseif (file_exists($video_site)) {
    /* Если директория /video существует, перенаправим туда и конец
скрипта*/
    header("Location: ../video/");
    exit;
}
/* Проверим, что конфигурация установлена в директории /sns */
elseif (file_exists($sns_site)) {
    /* Если директория /sns существует, перенаправим туда и конец скрипта*/
    header("Location: ../sns/");
    exit;
}
}
else {
    /* Если переменная $default_site равно "выбор в файле конфигуратора"
перенаправить на страницу, указанную в config.ini*/
    header("Location: ../" . "$default_site" . "/");
}
}
/* Если Переменные $file_config или file_exists не существуют,
то перенаправим на страницу установщика install.php */
else {
    header("Location: ../library/install/install.php");
}
?>
```

Обратите внимание на следующую часть кода:

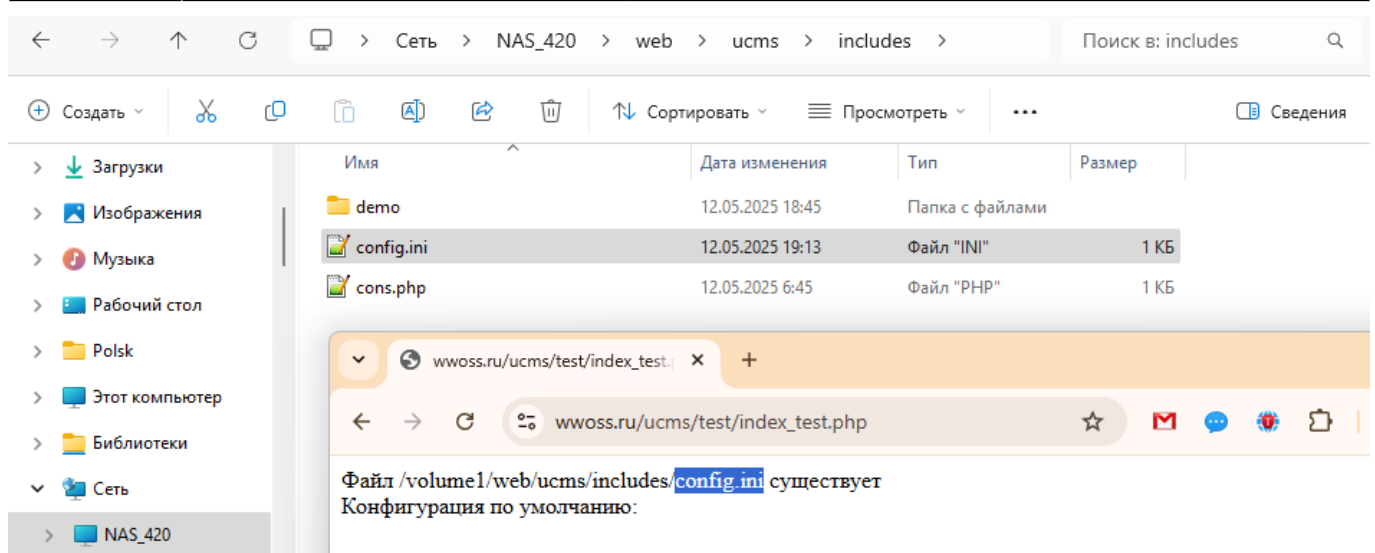
```
}
//exit;
/* зададим переменным файлы и директории */
```

- **exit;** необходима только лишь для остановки кода, чтобы увидеть проверку работы кода, если

в **config.ini** выбрана конфигурация по умолчанию.

Проверка существования директорий

Создаем в **/ucms/include** пустой файл **config.ini** и запускаем в браузере файл **/ucms/test/index_test.php**

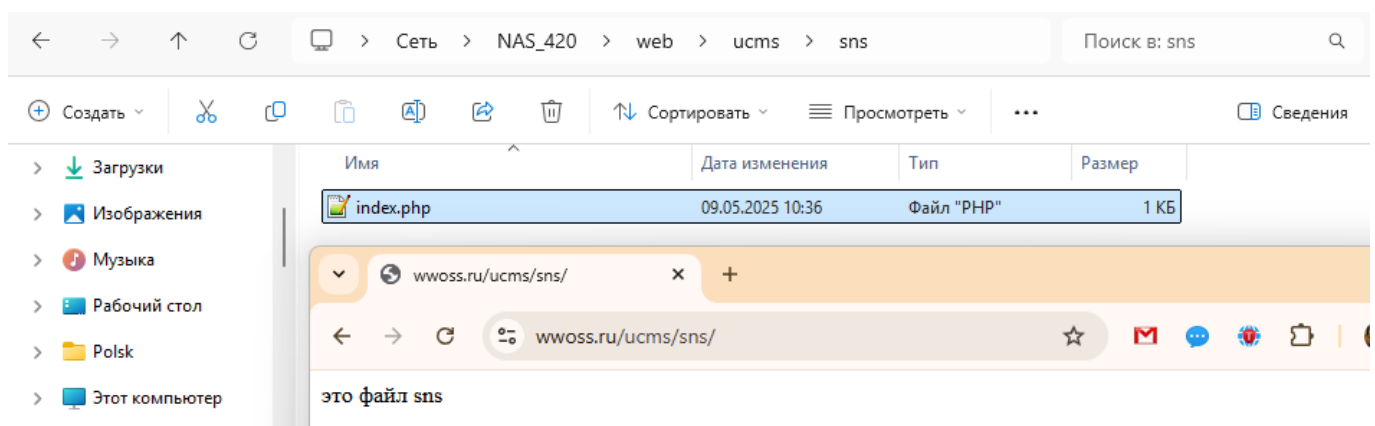


config.ini существует и он пустой, тогда:

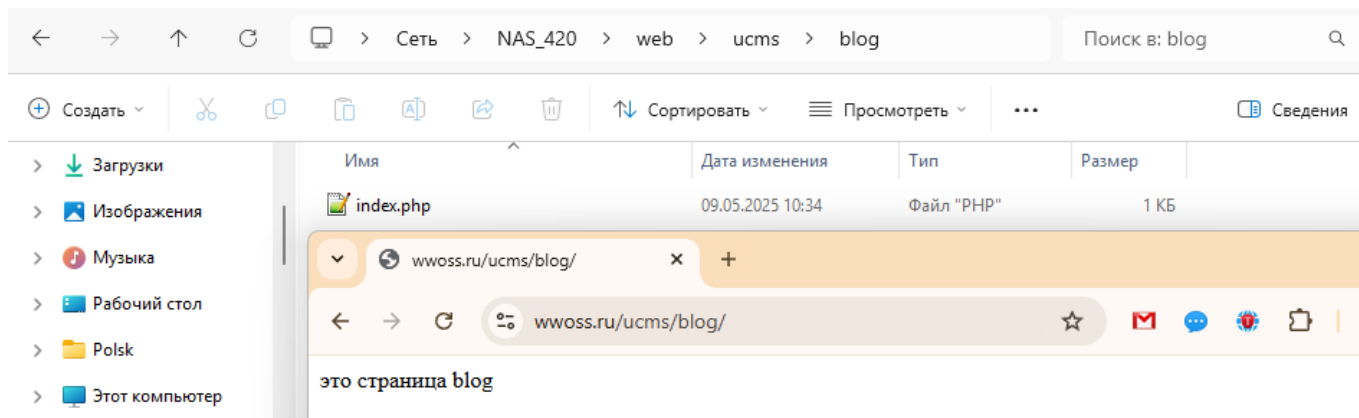
Проверим работу страницы **index_test.php** в веб - браузере, создав поочередно в директории **/ucms** вложенные директории: **bcw**, **blog**, **video**, и **sns** с вложенными файлами **index.php** внутри которых запишем имя директории. К примеру: создадим директорию **sns** с вложенным файлом **index.php** и записываем

`index.php`

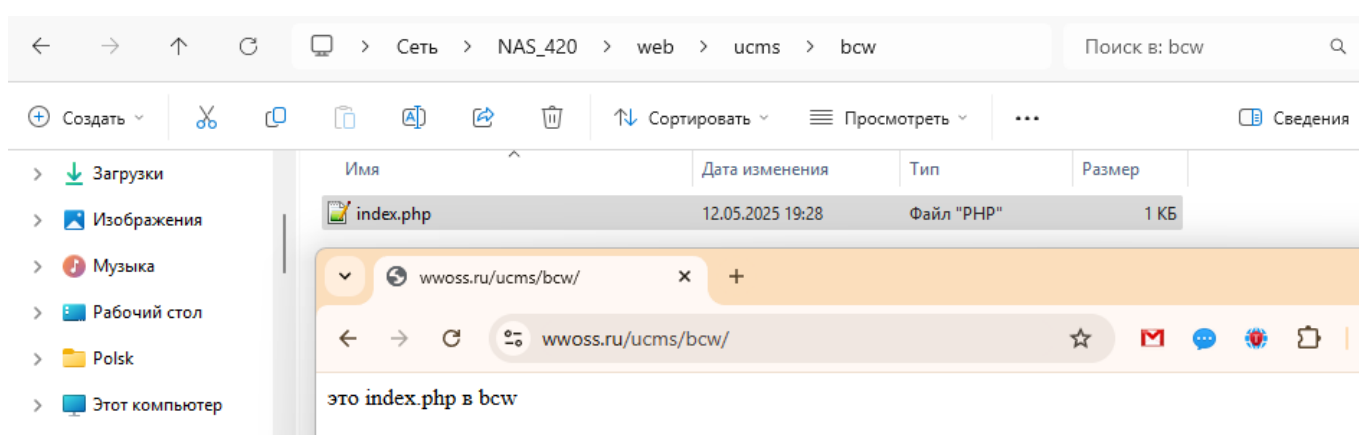
```
<? php
это файл sns
?>
```



А при добавленной директории **blog** переадресация на **/ucms/blog/** т.к. он раньше назначен у нас в проверке



И так поступаем с каждой директорией, проверяя (обновляя страница) логику поиска и открытия первой по алфавиту директории. Ну и при создании всех 4х директорий, мы получим первую по списку при запуске **index_test.php**



index_test_2.php

Создадим в директории **/test** файл **index_test_2.php** который будет нам создавать файл с записью конфигурации. К примеру: **video**

[index_test_2.php](#)

```
<!-- test/index_test_2.php -->

<?php
/* подключим файл с константами, для определения корневой директории приложения */
require_once __DIR__ . "../includes/cons.php";

/* Присваиваем переменной $file_config значение константы FOLDER
(а это директория ucms, записанная в файле cons.php) и добавляем
'/includes/config.php' - директорию includes и искомый файл
config.ini */
$file_config = FOLDER . '/includes/config.ini';
/* Если config.config.ini существует */
if (file_exists($file_config)) {
    /* выводим запись о существовании */
}
```

```
        echo "Файл $file_config существует". "<br/>";
/* Иначе выводим запись о существовании*/
    } else {
        echo "Файл $file_config не существует". "<br/>";
    }
$config_default_site = parse_ini_file(FOLDER .
"/includes/config.ini",true);
// Проверьте, успешно ли прочитан ini-файл.
// переменная $config не равно ЛОЖЬ
if ($config_default_site !== false) {
    // Доступ к индивидуальным настройкам

    $default_site =
$config_default_site['default_site']['default_site'];
    echo "Конфигурация по умолчанию: " . $default_site . "<br/>";
} else {
    echo "Ошибка чтения INI-файла.";
}
// Создаем массив пар
$databaseSections = [
    'default_site' => 'video',
];

// Записать конфигурацию в раздел «'default_site'»
$success = write_to_ini($databaseSections, 'default_site',
'../includes/config.ini');

// Проверим, успешно ли выполнена запись в файл.
if ($success) {
    echo "Запись в файл прошла успешно!";
} else {
    echo "Ошибка записи INI-файла.";
}
// Функция записи настроек в определенный раздел INI-файла
function write_to_ini($config, $section, $file) {
    $content = "\n; Application Configuration Settings\n\n[$section]\n;
Configuration\n";

    foreach ($config as $key => $value) {
        $content .= "$key = $value\n";
    }

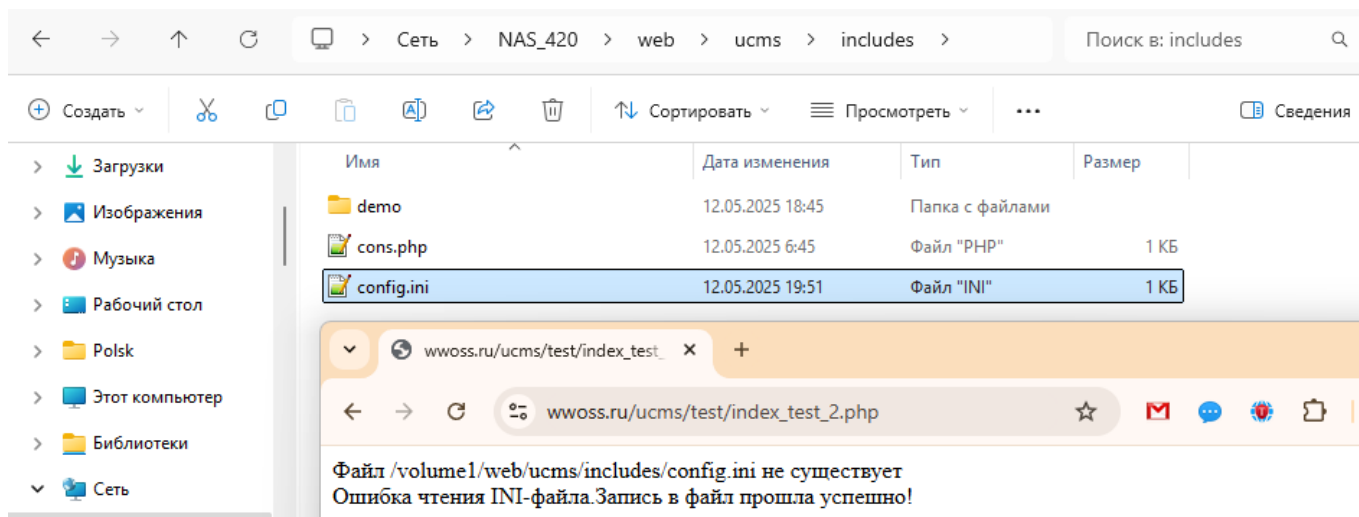
    // Добавить содержимое в файл
    return file_put_contents($file, $content, FILE_APPEND) !== false;
}
?>
```



Примечание:

Ранее **config.ini** создавали мы, значит и владелец файла на запись и чтение тоже мы, а нам необходимо передать права нашему приложению. Заодно проверим код записи в файл.

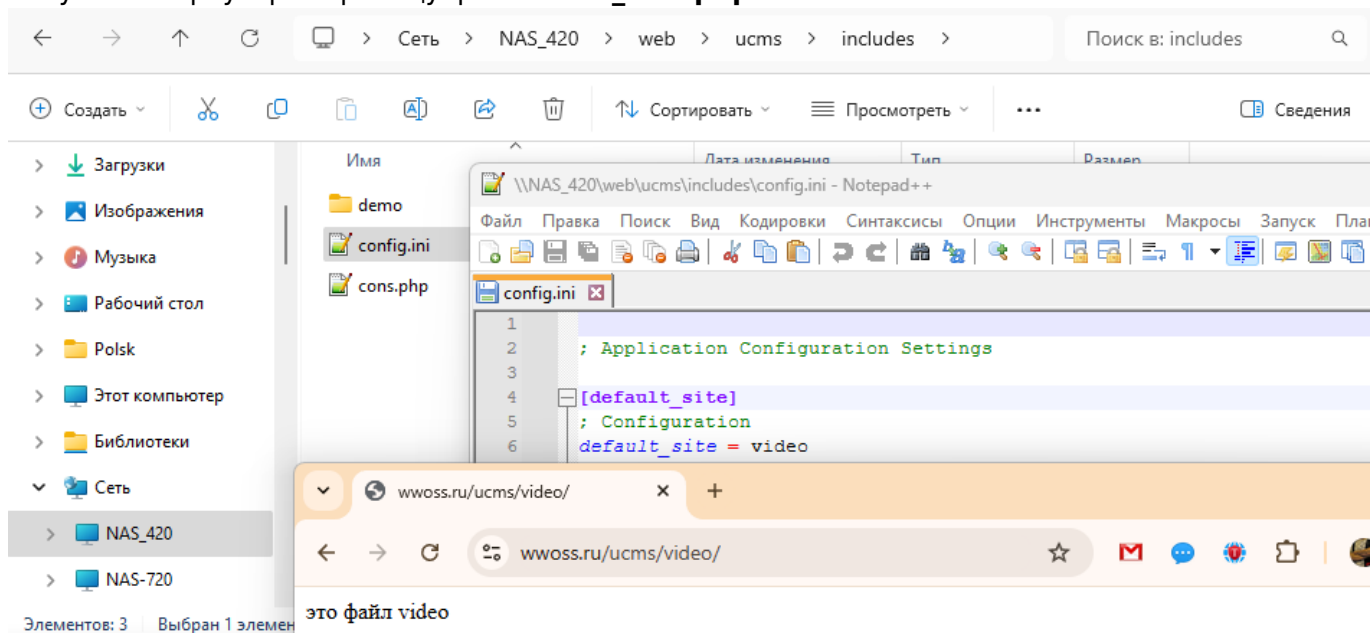
Удалим созданный нами файл **config.ini** и запустим в браузере файл **index_test_2.php**



т.к. мы удалили файл **config.ini** мы получили сообщение:

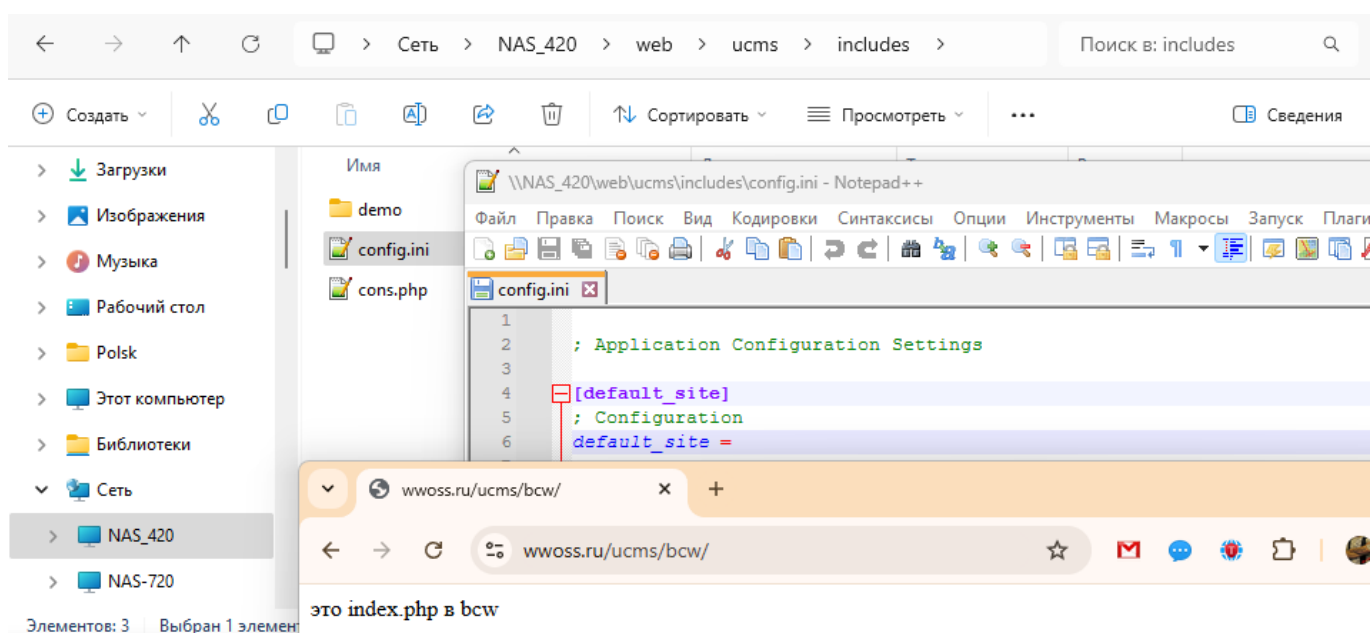
- **Файл /volume1/web/ucms/includes/config.ini** не существует - файла действительно нет
- **Ошибка чтения INI-файла.** - прочесть файл приложение не смогло
- **Запись в файл прошла успешно!** - был создан новый **config.ini** с записью **default_site = video**

Запустим в браузере страницу файл **index_test.php**



Так-же проверим работу логики при пустом значении переменной **\$default_site** в файле **config.ini** в директории **ucms/includes**

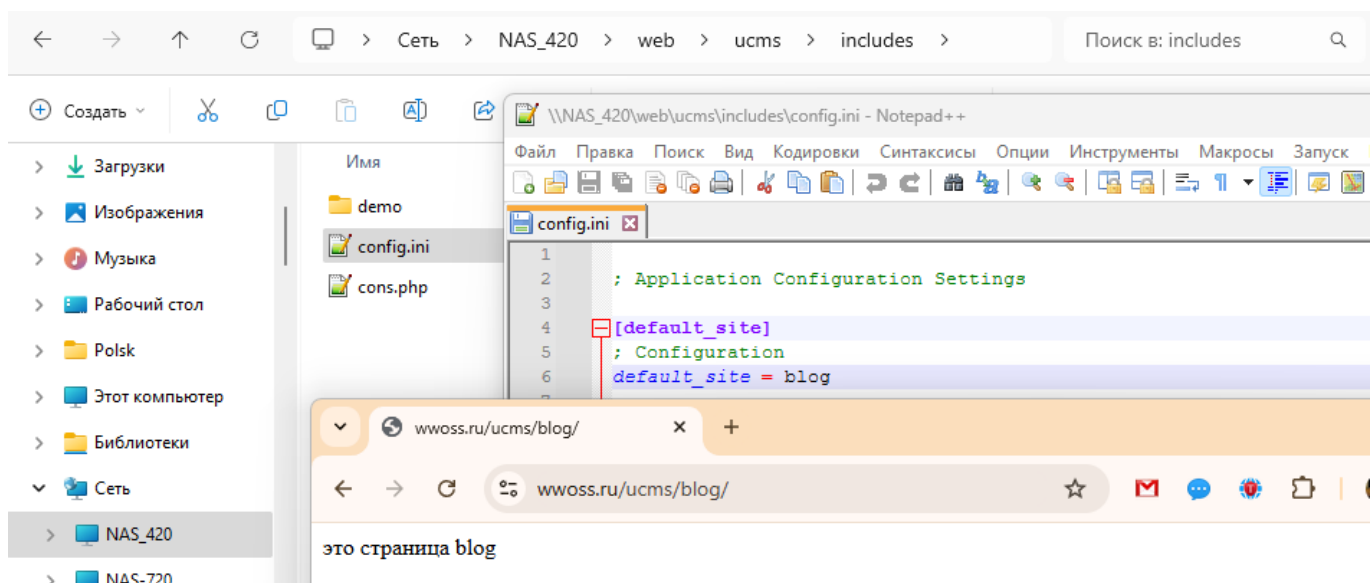
default_site =



Приложение перешло на первую по списку директорию.

Аналогично, для определенной переменной `$default_site`

default_site = blog



Логика index.php

- **ЕСЛИ** ядро приложение не установлено (т.е. отсутствуют файлы `version.php` и `config.ini` в директории `/includes`), то приложение перенаправит в `/library/install` к файлу `install.php`
- **ИНАЧЕ** проверит переменную `$default_site` в файле `config.ini`
- **ЕСЛИ** переменная `$default_site` = **пустое значение**, то приложение перенаправит на

индексный файл, первой из установленных конфигураций

- **ИНАЧЕ** получив значение переменной **\$default_site** из файла **config.ini** перенаправит к **index.php** установленной конфигурации, указанной пользователем.

По замыслу пользователь может установить несколько приложений и указать приложение по умолчанию в **config.ini**, тогда в корневой директории будет работать указанное приложение, а к остальным будет доступ через ссылку в коде программы на директорию другой конфигурации. К примеру:

Пользователь установил 2е конфигурации **blog** и **video** и выбрал основной конфигурацию **blog**, тогда в окне **index.php** будет работать конфигурация **blog**, а конфигурацию **video** будет доступна через указание директории **/video/video.php**

Файл index.php

Главным отличием файла **index.php** от **index_test.php** станут:

- корректировка директорий файлов
- вынос «логики» в основной файл функций **function.php** с созданием функции **check_start()**
- вывод конфигурации в «окне» файла **index.php** вместо вывода в отдельной директории

Создадим в директории **/ucms** файл входа в приложение **index.php**

[index.php](#)

```
<?php
/* index.php */
/* подключим файл с константами, для определения корневой
 *директории приложения и файлов конфигурации и функций */
include './includes/cons.php';
include FOLDER . '/includes/function.php';
/* запустим функцию проверки первого входа в приложение */
check_start();
?>
```

Создадим в директории **/includes** файл **function.php** и запишем в него нашу первую функцию **check_start()**;

[function.php](#)

```
<?php
/* Функция проверки установки ядра приложения и установленной конфигурации */
function check_start() {
    /* Присваиваем переменной $file_config значение константы FOLDER
    (а это директория ucms, записанная в файле cons.php) и добавляем
    '/includes/config.php' - директорию includes и искомый файл
    config.ini */
```

```
$file_config = FOLDER . '/includes/config.ini';

// Создание и вызов параметров из .ini файла
////////////////////////////////////
/* задаем переменной значение $config
 * parse_ini_file - обрабатывает конфигурационный файл
 * /includes/config.ini - путь к config.ini
 * ,true); - обрабатываем с секциями (зпголовками)
 */
$config_default_site = parse_ini_file(FOLDER .
"/includes/config.ini",true);
// Присвоим переменной $default_site значение из ini-файл.
$default_site = $config_default_site['default_site']['default_site'];
/* зададим переменным файлы и директории */
$file_config = FOLDER . '/includes/config.ini';
$file_version = FOLDER . '/version.txt';

$bcw_site = FOLDER . '/bcw';
$blog_site = FOLDER . '/blog';
$video_site = FOLDER . '/video';
$sns_site = FOLDER . '/sns';

/* Если переменные $file_config и file_exists существуют,
то проверим определение переменной конфигурации $default_site
по умолчанию в файле config.php */
if (file_exists($file_config) && file_exists($file_version)) {
    /* Если переменная $default_site равно "пустое место" */
    if ($default_site == ''){
        /* Проверим, что конфигурация установлена в директории /bcw */
        if (file_exists($bcw_site)) {
            /* Если директория /bcw существует, перенаправим туда и конец скрипта*/
            header("Location: ./bcw/");
            exit;
        }
        /* Проверим, что конфигурация установлена в директории /blog */
        elseif (file_exists($blog_site)) {
            /* Если директория /blog существует, перенаправим туда и конец скрипта*/
            header("Location: ./blog/");
            exit;
        }
        /* Проверим, что конфигурация установлена в директории /video */
        elseif (file_exists($video_site)) {
            /* Если директория /video существует, перенаправим туда и конец
скрипта*/
            header("Location: ./video/");
            exit;
        }
        /* Проверим, что конфигурация установлена в директории /sns */
        elseif (file_exists($sns_site)) {
            /* Если директория /sns существует, перенаправим туда и конец скрипта*/
```

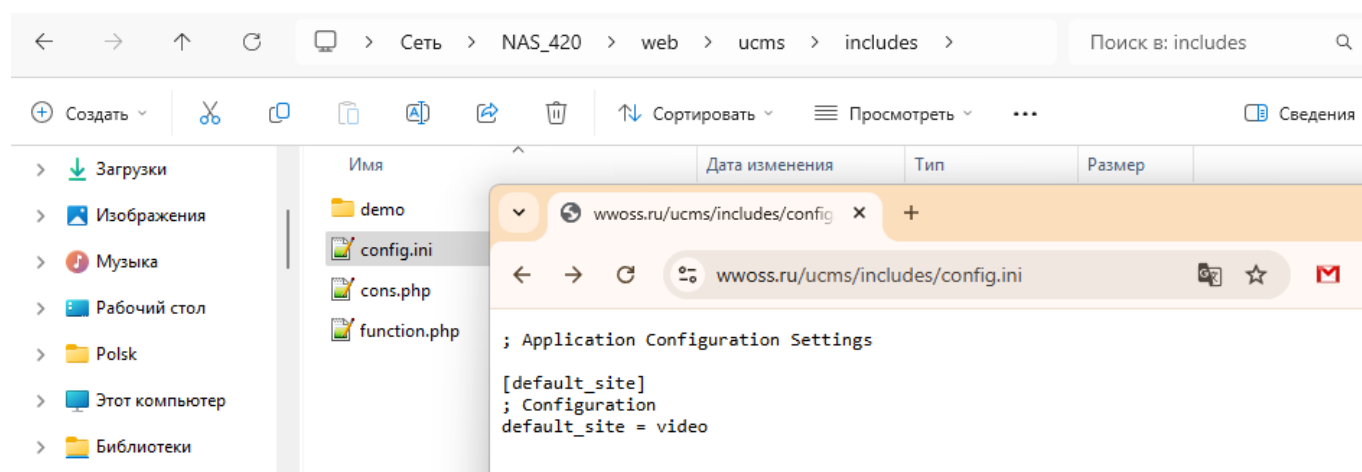
```

        header("Location: ./sns/");
        exit;
    }
}
else {
    /* Если переменная $default_site равно "выбор в файле конфигулятора"
    перенаправить на страницу, указанную в config.php*/
    header("Location: ./" . "$default_site" . "/");
    exit;
}
}

/* Если Переменные $file_config или file_exists не существуют,
то перенаправим на страницу установщика install.php */
else {
    header("Location: ./library/install/install.php");
    exit;
}
}
?>

```

Откроем в браузере наш файл **/ucms/includes/config.ini**



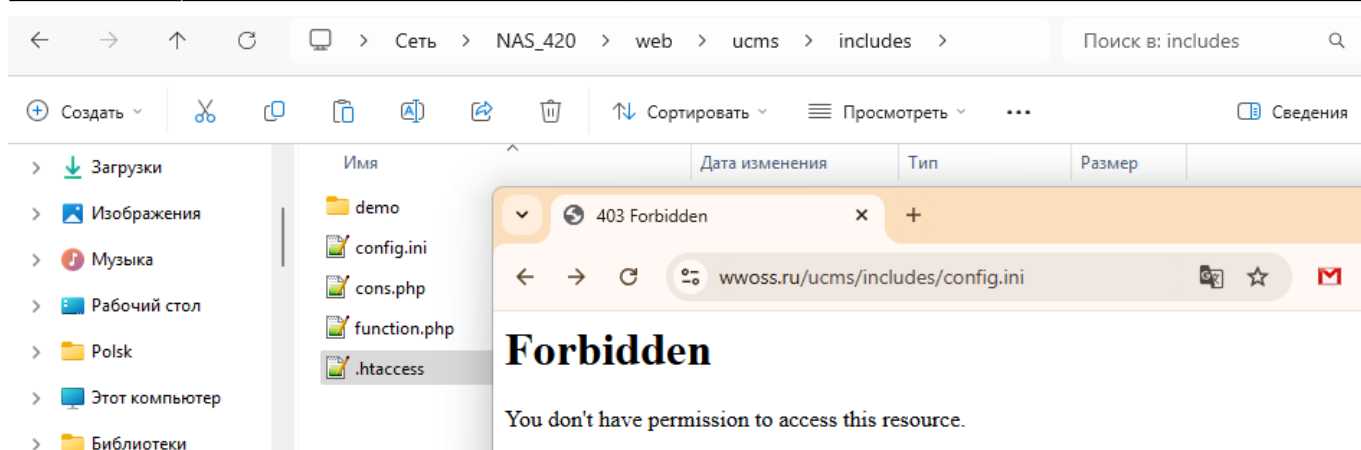
Как вы уже догадались мы закроем файл **config.ini** файлом **.htaccess**

.htaccess

```

# запретим просмотр файлов с расширением .ini следующих файлов
# (задается правило для перенаправления)
# config.ini
RewriteEngine On
RewriteRule ^(config)\.ini$ - [R=403,L,NC]

```



← 8.1. Создание директорий установщика 8.3. Файл lang.php →

From:
<https://wwoss.ru/> - **worldwide open-source software**

Permanent link:
<https://wwoss.ru/doku.php?id=software:development:demo:cms:ucms:index>

Last update: **2025/05/12 21:57**

