

**Мальчик Д.В.**

*студент фізико-математичного факультету*

**Карплюк С.О.**

*канд. пед. наук, доцент,*

*доцент кафедри прикладної математики та інформатики*

*Житомирський державний університет імені Івана Франка*

### **СТВОРЕННЯ WEB-САЙТУ З ВИКОРИСТАННЯМ PHP**

*Анотація. У даній статті будуть розглянуті переваги програмного забезпечення для створення різного роду проектів з використанням PHP, а також рекомендації для створення власного проекту.*

*Аннотация. В данной статье будут рассмотрены преимущества программного обеспечения для создания различного рода проектов с использованием PHP, а также рекомендации для создания собственного проекта.*

*Abstract. The benefits of software for creating various kinds of projects using PHP, and also the recommendations for creating your own project will be discussed in this article.*

В XXI столітті Web-технології дуже стрімко розвиваються. Користування мережею інтернет стало частиною нашого життя, тому пошук сайтів з приємним інтерфейсом та коректно поданою інформацією є доволі складним процесом для користувача. Для вирішення поставлених задач використовують галузь Web-програмування. Спеціалісти даної галузі можуть створити сайт за всіма правилами та побажаннями користувачів. Сучасні технології дозволяють кожній людині відкрити в собі Web-розробника. Технології HTML5 і PHP легкі у вивченні та подальшому використанні.

Для освоєння даних технологій Web-програмування потрібно вивчити мови розмітки сторінки, популярні стилі Web-сторінок, юзабіліті, кросбраузерність та вміти створювати інтерфейс, приємний для користувача.

Актуальність теми зумовлена тим, що Web-програмування стрімко розвивається та є популярним для розробників та користувачів.

Метою даної статті є поетапний огляд створення Web-сторінки.

PHP – це широко використовувана мова сценаріїв загального призначення з відкритим вихідним кодом, спеціально розроблений для написання

Web-додатків (сценаріїв), що виконуються на Web-сервері.

PHP дозволяє створювати якісні Web-додатки за дуже короткі терміни, отримуючи продукти, легко модифікуються і підтримувані в майбутньому.

Абревіатура PHP означає "Hypertext Preprocessor"(Препроцесор Гіпертексту). Синтаксис мови бере початок з C, Java і Perl. PHP досить простий для вивчення. Спочатку абревіатура PHP означала Preprocessor of Home Pages – препроцесор домашніх сторінок. Здебільшого його

синтаксис запозичений з таких мов, як C, Perl, Java, і при цьому додана масса можливостей, яких цим мовам бракує. Простіше кажучи, синтаксис PHP – це розумна альтернатива і строгості C, і "безмежності" Perl.

Основні засоби для розробки Web-додатків:

- автоматичне вилучення POST і GET-параметрів, а також змінних оточення Web-сервера в зумовлені масиви;
- взаємодія з великою кількістю різних систем управління базами даних (MySQL, MySQLi, SQLite, PostgreSQL, Oracle (OCI8), Oracle, Microsoft SQL Server, Sybase, ODBC, mSQL, IBM DB2, Cloudscape і Apache Derby, Informix, Ovrimos SQL, Lotus Notes , DB ++, DBM, dBase, DBX, FrontBase, FilePro, Ingres II, SESAM, Firebird / InterBase, Paradox File Access, MaxDB, Інтерфейс PDO);
- автоматизована відправка HTTP-заголовків;
- робота з HTTP-авторизацією;
- робота з cookies і сесіями;
- робота з локальними і віддаленими файлами, сокетамі;
- обробка файлів, що завантажуються на сервер;

Для початку роботи нам потрібно обрати операційну систему в якій ми будемо працювати. Але, коли йде річ про вибір операційної системи зазвичай розуміють операційні системи сімейства Unix(включаючи різновиди Linux) і операційної системи сімейства Windows. Але яку краще вибрати?

Хоча різниці між операційними системами сімейства Unix та Windows в контексті роботи з PHP несуттєва.

Якщо йде мова про створення не великого проекту то більшість надає перевагу Windows. Однак, якщо вирішили створювати професійний проект на PHP, вам слід зупинитися на операційних системах сімейства Unix, оскільки PHP розроблялося для роботи саме на Unix і саме тому, більшість бібліотек та доповнень вимагають саме Unix-подібних систем.

Далі нам потрібен Web-сервер. В залежності від операційної системи, він може бути наприклад Apache в Linux і MacOS X; IIS в Windows. Також можемо орендувати Web-сервер або деякий дисковий простір на Web - сервері. У такому випадку, не потрібно встановлювати додаткове програмне забезпечення.

Також нам потрібен сервер баз даних. В даний час частіше використовується MySQL разом з PHP. Сервер баз даних MySQL має хороші характеристики, розповсюджується безкоштовно і має найбільшу поширеність у взаємодії саме з PHP. Є ще сервер баз даних PostgreSQL, який останнім часом набирає популярності.

А тепер алгоритму за яким будемо будувати наш сайт.

#### Постановка головного завдання

Архітектура розробляється зверху вниз. Це означає, що спершу ви продумуєте найголовніші підсистеми, перш ніж приступити до обслуговуючим їх або більше технічним. Таким методом ми позбавимо себе від костилі в коді та постійного переписування коду. Це не єдиний метод, але є одним з найкращих для новачків. А далі ми повинні зробити вибір, що на нашому сайті буде головним. Для блогу – це записи, фірми – це послуги.

Далі ми можемо написати частину

```
код:
```

```
class записи
{
}
}
```

Це наша головна підсистема, зразу підпишемо чим вона буде управляти.

Функції нашої системи.

Далі ми повинні на нашій підсистемі прописати дії які вона буде

виконувати.

Приклад:

```
class записи
{
    public function создать_запись() {}
    public function удалить_запись() {}
    public function получить_запись() {}
}
```

Зрозуміти з чим маємо справу.

Для нас, як програмістів – це просто сукупність даних. У третьому кроці визначаємо, що це за сукупність.

Після чого ми повинні створити унікальний код для наших записів, товарів, послуг за допомогою якого комп'ютер буде знаходити потрібну нам інформацію. Це або порядковий номер (id), унікальний для кожного запису або якесь, настільки ж унікальне, строкове значення.

Визначення необхідних знань.

Далі ми повинні написати, що повинна знати система щоб виконати певну дію. Для дії "створити запис" потрібно знати заголовок і текст запису.

Приклад коду:

```
class записи
{
    public function создать_запись($заголовок, $текст) {}
    public function удалить_запись($идентификатор) {}
    public function получить_запись($идентификатор) {}
}
```

Те, що вийшло називається інтерфейсом вашого класу / модуля / підсистеми.

Визначення результатів.

Якщо у комп'ютера все вийшло, то:

1. створити запис: буде створений новий запис, а ми отримаємо її ідентифікатор;

2. видалити запис: буде видалена конкретний запис, а ми отримаємо повідомлення "готово";
3. отримати запис: одержимо потрібний запис;

Якщо у комп'ютера чогось не вийшло, то:

1. створити запис: буде виданий виняток;
2. видалити запис: буде виданий виняток;
3. отримати запис: буде виданий виняток.

Для ситуацій, коли у комп'ютера сталася помилка при виконанні дії потрібно зробити однакову поведінку у всіх модулях / класах / підсистемах нашої програми. PHP надає всього два варіанти: `trigger_error` і `throw new Exception`.

Аналогічні дії виконуємо заново. Повторюємо 1-5 кроки для головних підсистем які будуть використовуватися на нашому сайті. Після цього, пишемо код. Коли ви створите все що вам потрібно можна почати кодувати.

Отже, програмне забезпечення для створення різного роду проектів з використанням PHP, а також рекомендації для поетапного створення власного проекту будуть доцільними для людей які тільки починають розробляти власні проекти на мові Web-розробки PHP.

### **СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ:**

1. Что такое PHP? [Електронний ресурс] – Режим доступу : <http://www.php.su/php/?php>
2. Программное обеспечение для PHP [Електронний ресурс] – Режим доступу: <http://www.php.su/faq/?software>
3. Общие инструкции по установке [Електронний ресурс] – Режим доступу: <http://php.net/manual/ru/install.general.php>
4. Пошаговый алгоритм создания архитектуры PHP-сайта [Електронний ресурс] – Режим доступу : <http://habrahabr.ru/post/247495/>