

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ**

**КАФЕДРА КОМП'ЮТЕРНИХ НАУК**

# **КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА**

**на тему:**

**«Учбово-методичний ресурс для навчання  
студентів»**

**Завідувач  
випускаючої кафедри**

**Довбиш А.С.**

**Керівник роботи**

**Шелехов І.В.**

**Студента групи ІК.м – 91**

**Левченко А.Є.**

**СУМИ 2020**

Факультет \_\_\_\_\_ *ЕЛІТ* \_\_\_\_\_ Кафедра \_\_\_\_\_ *Комп'ютерних наук* \_\_\_\_\_

Спеціальність \_\_\_\_\_ *Інформаційно-комунікаційні технології* \_\_\_\_\_

Затверджую:

Зав.кафедрою \_\_\_\_\_

“ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_р.

## **ЗАВДАННЯ**

### **НА КВАЛІФІКАЦІЙНУ МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТОВІ**

*Левченка Андрія Євгеновича*

---

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) \_\_\_\_\_

затверджую наказом по інституту від “ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ р. № \_\_\_\_\_

2. Термін здачі студентом закінченого проекту (роботи)

\_\_\_\_\_

3. Вхідні дані до проекту роботи

\_\_\_\_\_  
\_\_\_\_\_

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)

*1) Інформаційний огляд 2) Вибір методу рішення 3) Проектування та програмне забезпечення системи. 4) Розробка інформаційної системи*

---

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання \_\_\_\_\_

Керівник

\_\_\_\_\_  
(підпис)

Завдання прийняв до виконання

\_\_\_\_\_  
(підпис)

### КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної магістерської роботи	Термін виконання проекту (роботи)	Примітка
1.			
2.			
3.			
4.			

Студент – дипломник

\_\_\_\_\_  
(підпис)

Керівник проекту

\_\_\_\_\_  
(підпис)

## РЕФЕРАТ

**Записка:** стор. – 66, рис. – 31, додатків – 1, джерел – 11

**Об'єкт дослідження:** Учбово-методичний ресурс для навчання студентів.

**Мета роботи:** розробка Учбово-методичний ресурс для навчання студентів.

**Методи досліджень:** в процесі виконання програмної реалізації було застосовано комбінацію технологій такі, як: HTML, CSS, JavaScript, Bootstrap, jQuery, PHP, MySQL.

**Результати:** були проаналізовані альтернативні версії учбових веб-ресурсів на ринку, їх плюси та мінуси. Також був проведений аналіз по вибору методів реалізації даної роботи.

ВЕБ-РЕСУР, CMS, УЧБОВИЙ ДОДАТОК HTML, CSS, СКРИПТИ,  
БІБЛІОТЕКА, ФРЕЙМБОРК, ХАМРР, PHP, JAVASCRIPT, BOOTSTRAP,

## ЗМІСТ

ВСТУП.....	5
1. ІНФОРМАЦІЙНИЙ ОГЛЯД .....	7
1.1 Загальний аналіз систем управління навчання.....	7
1.2 Переваги та недоліки дистанційного навчання.....	11
1.3 Технології які використовуються при розробці навчального веб-ресурса .....	15
1.4 Огляд існуючих учбових веб-ресурсів.....	16
1.5 Постановка задачі.....	17
2. ВИБІР МЕТОДУ РІШЕННЯ ПОСТАВЛЕННОЇ ЗАДАЧІ.....	18
2.1 Вибір системи керування вмістом CMS та установка.....	18
2.2 Вибір технологій .....	23
2.3 Вибір структури патерну.....	31
2.4 Вибір та установка локального серверу .....	32
ВИСНОВОКИ .....	37
СПИСОК ЛІТЕРАТУРИ .....	55

## ВСТУП

З розвитком суспільства зростає попит на якісні освітні послуги, що дозволяють підвищувати кваліфікацію без відриву від основної діяльності. У наш час недостатньо здобути освіту в молоді роки, необхідно підвищувати свій освітній рівень протягом всього активного життя. Комп'ютери та комп'ютерні мережі дозволяють створювати високоефективні засоби навчання в освіті. В основі таких систем лежать інформаційні бази даних, головну частину яких складають електронні засоби навчання.

Розробка електронних засобів навчання, є досить складною справою. Для створення ефективних засобів необхідно раціональне поєднання всіх можливостей наданих сучасним комп'ютером: гіпертексту, графіки, відео, звуку і анімації.

Основна маса вже створених електронних засобів навчання - це навчальні матеріали для студентів і школярів, навчальні матеріали по комп'ютерним програмам, різні енциклопедії та довідники. Існуючі електронні засоби навчання можна розділити на наступні категорії:

- електронні версії (копії) паперових підручників;
- оцифровані відеозаписи лекцій;
- електронні версії (копії) довідників, словників та енциклопедій;
- мультимедійні електронні навчальні посібники.

Електронні навчальні засоби навчання - комп'ютерні, педагогічні програмні засоби, призначені для надання нової інформації, що доповнює друковані видання для особистого вивчення та перевіряє знання та вміння, набуті студентами, в обмеженій формі.

Електронні навчальні засоби навчання зручні тим, що досвідчений користувач (розробник) при знанні синтаксису, використовуваних при розробці засобів і має достатню для цього кваліфікацію, може в будь-який момент

зробити модифікацію матеріалу, що міститься в електронних засобах навчання. Це обумовлено тим, що періодично змінюються стандарти і вимоги до освітнього матеріалу. Наприклад, йде скорочення або ж збільшення лекційного матеріалу з певних тем.

Тому метою магістерської роботи є розробка учбово-методичного ресурсу для навчання студентів. Це система управління навчання де студенти знайдуть різноманітні програми, курсів та тестувань для підвищення рівня своїх знань.

# 1. ІНФОРМАЦІЙНИЙ ОГЛЯД

## 1.1 Загальний аналіз систем управління навчання

Історія виникнення електронного навчання почалася на початку 20 століття зі створення аналогового комп'ютера для симуляції польотів на літаках Сполучених Штатів Америки. Створення персональних комп'ютерів зробило величезний вплив на виток розвитку програмного забезпечення, і до середини 80-х років доступність персональних комп'ютерів в світі досягла такої точки, коли почали створюватися компанії, що спеціалізуються на освітньому програмному забезпеченні. У світі, починаючи з середини 90-х років, кількість розроблених програм в цій сфері налічує тисячі.

Історія українського ринку бере початок з 90-х років, коли фахівці в області комп'ютерних технологій розробляли перші програмні продукти, що дозволяють здійснювати взаємодію між студентами та викладачами на відстані. Після прийняття концепції створення та розвитку системи дистанційної освіти в 1992 році, в декількох великих вузів країни були впроваджені перші прототипи систем дистанційної освіти. Однак на відміну від світових технологій, в Україні рівень комп'ютеризації був настільки малий, що електронну освіту швидше було вважати за вид заочного навчання.

Розвитку електронного навчання сприяло формування ринку бізнес-освіти. Організації, що надають послуги в області бізнес-навчання, оцінивши ефективність нового способу, стали поступово переходити на електронний варіант.

На сьогоднішній день в світі електронна освіта використовується повсюдно. Наприклад, в США вже більше 90% вузів та шкіл, а також компаній, що мають чисельність понад тисячу - півтори тисячі осіб, використовують цю форму навчання. У порівнянні з ситуацією в світі, розвиток ринку електронного навчання в Україні, за оцінками фахівців, відстає на 5-7 років. [1]



Електронне навчання можна умовно розділити на 2 сегменти:

- Компанії-розробники програмного забезпечення;
- Компанії-розробники електронних курсів і навчального контенту.

Перший сегмент становить основну частину ринку послуг електронного навчання. До нього відносяться компанії-розробники програмного забезпечення, яких також можна умовно розділити на сегменти.

- Розробка LMS (Learning management system - система управління навчанням) і LCMS (Learning Content Management Systems - система управління навчальним контентом).
- Інструменти для організації і проведення вебінарів або віртуальних класів;
- Створення засобів для розробки електронних курсів і навчального контенту (системи розробки курсів).

Поділ на сегменти умовне, так як багато компаній представляють всі сегменти e-learning, інші - мають розробки в двох або одному сегменті.

- LMS являє собою платформу для розгортання електронного навчання.

Платформа виконує наступні функції:

- надає інтерфейс для процесу навчання і взаємодії викладача і учнів,
- надає навчаються доступ до навчального порталу, який є відповідною точкою для доставки всього навчального контенту;
- надає необхідні інструменти для формування навчальних програм,
- контролю їх проходження, складання звітів про результативність навчання,
- організації комунікацій між студентами і викладачами;
- керує учнями, включаючи завдання реєстрації і контролю доступу користувачів до системи і до навчального контенту;

- організовує слухачів в групи для надання їм загальних курсів і складання звітності;
- керує аудиторними і викладацькими ресурсами;
- дає інтеграцію додатковим елементам для навчального процесу (лабораторні роботи, практичні заняття, тести та засоби спільної роботи);
- забезпечує механізми захисту, необхідні для мережевого середовища e-learning

Дуже часто термін LMS використовують як синонім терміну система дистанційного навчання. Однак це не зовсім так. LMS - це лише програмне забезпечення для функціонування самої системи дистанційного навчання. Сама СДО включає в себе технологічний аспект (програмне забезпечення та апаратну частину), процеси організації навчання в умовах ДО, і людей, які будуть забезпечувати ці процеси (адміністратори, викладачі, методисти, студенти, абітурієнти).

LCMS - це система, яка сконцентрована на завданні управління змістом навчального контенту, але не самим процесом навчання. Сама система орієнтована не на кінцевих користувачів e-learning, тобто навчаючихся, а на фахівців з розробки електронних курсів, тобто програмістів. LCMS орієнтована на розвиток, управління і публікацію вмісту навчального контенту і поставляється, як правило, через LMS. LCMS є багатокористувацької середовищем, в якій розробники можуть створювати і зберігати навчальний контент, керуючи їм за допомогою центрального сховища об'єктів. У цьому відмінність LMS і LCMS – LMS не має можливості створення і зміни курсів, вона не може використовувати вміст одного курсу для створення іншого. Це входить в можливості LCMS – система може створювати, управляти і поставляти не тільки окремі навчальні модулі, а й управляти і редагувати окремі частини навчального контенту. Система управління навчальним контентом зберігає матеріали навчального курсу, створює класифікацію

матеріалів, веде історію змін контенту, дозволяє вести колективну роботу над наповненням курсу і формувати його з окремих компонентів. Ці матеріали можуть включати в себе мультимедійні дані, створені в інших засобах розробки, моделювання, текст, графіку або будь-який інший об'єкт, що становить вміст контенту.

Другий сегмент становить невелику частку ринку e-learning. Компанії, що займаються створенням електронних курсів і навчального контенту, виконують безпосередньо наповнення електронного курсу базою знань для учнів, тобто створюють архів навчальних матеріалів, лекцій, створюють тести і опитування і т.п.

Основними формами навчального контенту є електронний курс симуляція (віртуальне середовище, що імітує реальні умови діяльності), лекція (веб семінар, запис вебінару). Може використовуватися практично будь-який електронний контент: графіка, анімація, відео, документи і т.д.

Розробка навчального контенту може бути колективною і індивідуальною, на основі готових шаблонів або повністю авторської.

## 1.2 Переваги та недоліки дистанційного навчання

З кожним роком число людей все зростає та має необхідність в навчанні певного типу А зовнішні навчальні заклади мають можливість працювати на повну потужність. Вирішенням цієї ситуації є дистанційне навчання.

Очевидними перевагами дистанційного навчання є:

- **Дистанційність.** Той, кого навчають має можливість навчатися, незалежно від відстані до університету.
- **Гнучкість.** Студент не має будь-яких обмежень за часом і займається в зручні для себе час і в оптимальному темпі, замість того, щоб відвідувати заняття у встановлений час. Але при всьому при цьому, студент має чіткий навчальний план, виконання якого строго контролюється викладачем.
- **Рівнобіжність.** Навчання може проводитися разом з основною діяльністю і роботою людини, тобто "не відриваючись від виробництва".
- **Асинхронність.** Тобто, в процесі навчання керівник і учень працюють за зручним для кожного розкладом.

Дистанційне навчання дозволяє забезпечити максимально особистісний підхід з боку керівника. Викладач курсів може знаходитися в постійному інтерактивному контакті зі слухачами: фіксувати відвідуваність, перевіряти завдання, контрольні роботи та тести, обговорювати на вебінарах, форумах і в чатах різні проблеми, відповідати на питання, індивідуально пояснювати найбільш складні теми.

В даний час сформувався неоднозначний підхід і оцінка можливостей дистанційного навчання. Багато хто скептично ставиться до дистанційного навчання, вважаючи його якість низьким і не заслуговує на увагу. Саме такий підхід іноді породжує не зовсім раціональну оцінку.

- Дистанційне навчання не заслуговує на довіру.

Якщо більшість соціальних фінансових установ підприємств та державних установ бояться впровадження нових технологій, вони будуть розтоптані. Справа в тому, що студентам не потрібне інше програмне забезпечення для успішного дистанційного навчання. Все, що треба - це комп'ютер, підключений до інтернету. Тут немає складних інструментів і цінного програмного забезпечення. З технічної точки зору, електронне навчання - це дуже простий процес, особливо тому, що більшість людей мають досить надійні комп'ютери та постійний доступ до мережі інтернета.

- Студенти не мають змоги групової взаємодії.

В останній час рівень соціальних взаємодій між людьми зашкалює. Різноманітні академічні платформи електронного навчання використовують швидке зростання соціальних мереж, такі як Twitter та Facebook, що дають людям можливість для спілкування та взаємодії, таким чином створювати сприятливі умови для навчання.

На відміну від класичних навчальних закладів, де студенти обмежені за стінами аудиторії, ті, хто вибрали дистанційну форму навчання, мають можливість спілкуватися з людьми зі всього світу.

- Студенти гірше навчаються, так як не відчують контроль з боку викладача над своєю роботою.

Тобто якщо студентка або студент не зацікавлений в нових знаннях, їй або йому все одно: чи спостерігає за його роботою викладач або ні. В тому часі як електронне навчання не передбачає, що студент повинен у певний час в певному місці, при цьому забезпечуючи йому доступ до вчителя і іншим однокурсникам, що здатні надати допомогу і підтримку.

Нерозумно звинувачувати все в навчанні в інтернеті. Як спосіб передачі інформації, електронне навчання не несе ніякої відповідальності за шкідливі звички учнів.

- Дистанційне навчання писивне.

Інколи в аудиторії є студенти більш пасивні в порівнянні з тими, хто проходить курс дистанційного навчання. В останньому випадку студенти, які присутні безпосередньо на лекції, будуть більше залучені до процесу, оскільки вони фізично взаємодіють з викладачем.

- Електронне навчання – нудота.

Якщо онлайн-навчання охрестили “нудним”, це тільки тому, що викладач погано запланував роботу. Нудно на лекції на дистанційній формі навчання не буває: як і в звичайному класі, ви можете спілкуватися з іншими студентами на вебінарах, піднімати руки, задавати питання, спілкуватися з викладачами та використовувати Інтернет для проведення досліджень та лекцій.

Але також є і очевидні недоліки даної форми навчання.

- Деякі спеціальності можна освоїти дистанційно.

Медичні, та деякі технічні спеціальності, які вимагають великого обсягу практичних занять для набуття професійних навичок вкрай важко, часом і неможливо отримати дистанційно.

- Відсутність реального, «людського» спілкування між учнями та викладачами.

Тобто відсутні всі ті моменти, пов'язані з індивідуальним підходом до навчання і вихованням. А якщо поруч немає викладача, який зазвичай емоційно забарвлює знання і сприяє сприйняттю матеріалу, це, звичайно, значний мінус.

- Обмеженість можливостей практичних занять.

Відсутність деяких можливостей практичних занять, необхідних для закріплення теорії і більш якісного засвоєння знань, відсутня в цій формі навчання.

- Якість дистанційних курсів.

Навчальні електронні програми та курси не завжди добре розроблені і відповідають усім вимогам через недостатню кваліфікацію фахівців, що створюють подібні навчальні посібники, так як на сьогоднішній день це ще нове і недостатньо вивчене напрямом.

Якщо підвести підсумок, то дистанційна освіта - річ досить зручна і корисна. Безсумнівно, це крок в освіту майбутнього. Але все-таки не варто забувати традиційну освіту, яке дасть нам більше навичок, особливо при отриманні основної освіти. Дистанційне навчання поки рекомендується отримувати, як додаткове або якщо у людини немає можливості вчитися традиційно з якихось вагомих причин.

А ось в якості додаткової освіти або підвищення професійної кваліфікації дистанційне навчання лідирує, тому що «учень» вже має професійні навички в області досліджуваного предмета і має можливість продовжувати освіту без відриву від роботи.[1]

### 1.3 Технології які використовуються при розробці навчального веб-ресурса

Для реалізації клієнтської частини веб-додатків сьогодні найчастіше використовуються наступні технології:

- HTML , XHTML
- CSS
- ActiveX
- Adobe Flash , Adobe Flex
- Java
- JavaScript
- Silverlight
- Ajax
- JQuery

Для розробки серверної частини можуть бути використовувані такі технології:

<u>Назва</u>	<u>Ліцензія</u>	<u>Веб-сервер</u>
ASP	<u>пропрієтарна</u>	спеціалізований
ASP.NET	<u>пропрієтарна</u>	спеціалізований
<u>Java</u>	вільна	безліч, в тому числі вільних
PHP	вільна	практично будь-який
<u>Python</u>	вільна	практично будь-який
<u>Ruby</u>	вільна	практично будь-який

Таблиця 1.3.1 — Технології та мови програмування, які використовуються на сервері



## 1.4 Огляд існуючих учбових веб-ресурсів

В останні кілька років термін "дистанційна освіта" повсюдно замінюється на «електронне навчання» (E-learning), що означає процес навчання в електронних формі, через мережу інтернет або інтранет з використанням систем управління навчанням.

Тому розглянемо на прикладі існуючі учбові веб-ресурси.

- **INTUIT** – організація, що надає за допомогою власного сайту послуги дистанційного навчання за кількома освітніми програмами, багато з яких стосуються інформаційних технологій.
- **Coursera** – унікальна міжнародна платформа дистанційної освіти з курсами кращих університетів усього світу.

Що пропонується: більше 2,000 онлайн-курсів 146 навчальних закладів, включаючи Манчестерський університет, Принстонський університет, Єльський університет і Стенфордський університет/

- **Lingualmeo** – освітня платформа для вивчення і практики іноземної мови, побудована на ігровій механіці.
- **Codeacademy** – безкоштовні інтерактивні завдання та практика для навчання «з нуля» в області створення веб-сайтів, розробки додатків і ігор, HTML & CSS, JavaScript, jQuery та унікальні повністю безкоштовні курси різної спрямованості, завдання та лекції, розроблені провідними американськими професіоналами в сфері програмного забезпечення.
- **Stepic** – російськомовні і англійськомовні курси в галузі гуманітарних, точних і природничих наук, інформатики та статистики. Постійно поповнюється каталог навчальних програм; університетські і авторські курси від експертів; зручний російськомовний інтерфейс і мобільний додаток в AppStore.
- **Open Yale Courses** – безкоштовні відео-лекції та інші навчальні матеріали курсів Єльського університету в різних сферах, від хімії та

економіки до англійської мови, історії та музики безкоштовні і якісні академічні ресурси; можливість оцінити свій рівень володіння англійською мовою та підготовки до навчання в англійськомовному вузі.

Мій проект буде спеціалізуватися на наданні курсів по різних учбових програмах та тестування користувача, завдяки яким він зможе підвищити свій рівень теоретичних і практичних знань.

### **1.5 Постановка задачі**

Постанова задачі даного проекту є створення учбово методичного ресурсу для навчання студентів. Тому актуальним є розробити учбово-методичний матеріал з тестуванням, в основі якого лежатиме інтернет сайт.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- Обрати CMS для створення сайту для дистанційного засобу навчання та встановити.
- Розробити структуру та дизайн сайту інтерфейс системи повинен бути доступний будь-якому користувачеві.
- Встановлення локального серверу.
- Розробити теоретичний і практичний матеріал сайту, а так само створити методичні рекомендації щодо використання сайту.
- Розробити систему тестів та її оцінювання.
- Форма реєстрації для проходження матеріалу.
- Пошук за назвою структурного підрозділу.
- Зворотній зв'язок.
- Виконати тестування результату.

## 2. ВИБІР МЕТОДУ РІШЕННЯ ПОСТАВЛЕННОЇ ЗАДАЧІ

### 2.1 Вибір системи керування вмістом CMS та установка

Система управління вмістом (контентом) (англ. Content management system, CMS) - це програмне забезпечення що використовується для створення веб-сайтів та створення вмісту для публікації в інтернеті.

Основні функції CMS:

- Легкий спосіб створювати, редагувати, публікувати та керувати веб-вмістом (включаючи текст, зображення, відео та аудіо).
- Найважливішою інтеграцією є можливість підключення персоналізованої маркетингової презентації до ваших систем цифрової комерції.
- подання інформації у зручному вигляді для навігації та пошуку.

Система керування контентом можна знайти найрізноманітніші дані: фільми, документи, фотографії, наукові дані номерів телефонів і тд. Подібна система часто використовується для управління, зберігання, публікації документації та його перегляду.

Найголовніша функція CMS - це відображувати сторінки сайту користувачеві, формуючи їх вміст «на льоту» із заздалегідь визначених шаблонів з дизайном і контенту, мається на увазі малюнків, текстів, таблиць та інших матеріалів, що збережені в базі даних, створення сторінки користувачу в момент його запиту. У цьому випадку, залежно від ситуації, користувачеві може бути показана якась унікальна інформація, яку інші люди не побачать. Наприклад, вміст його кошика в інтернет-магазині.

При цьому CMS не просто генерує сторінки користувачам, а намагається зробити це якомога швидше, щоб максимальна кількість були оброблені запити за одиницю часу. Також захищає дані від недобросовісних користувачів.

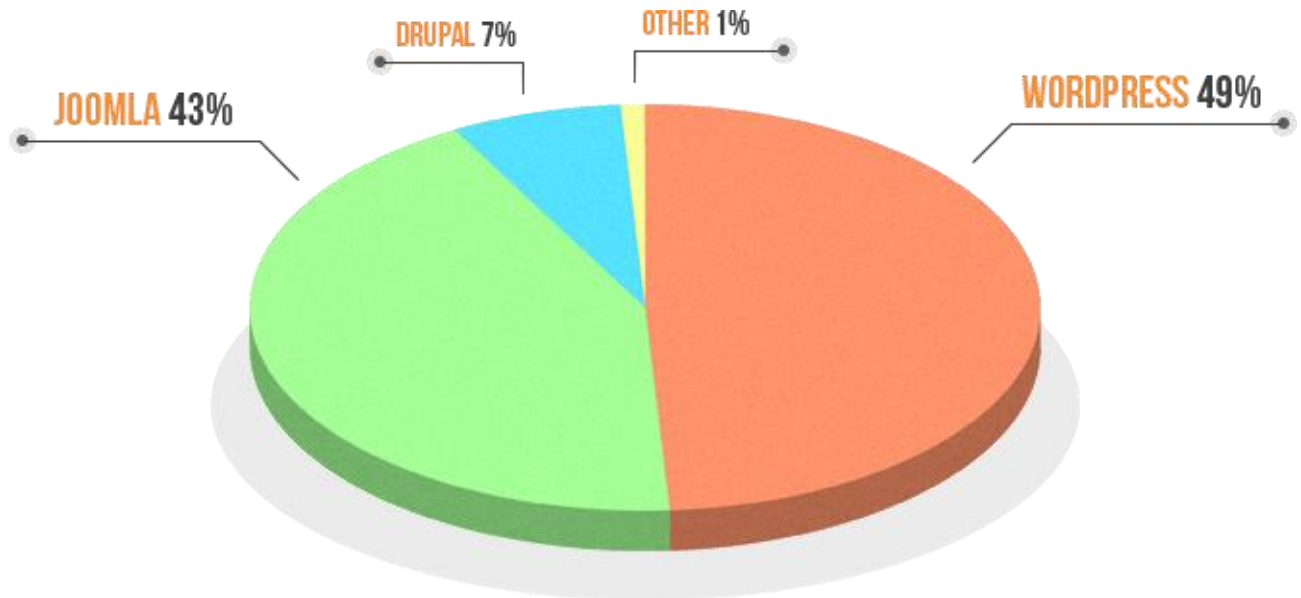


Рисунок 2.1.1 — Найпопулярніші системи керування вмістом за 2019р.

**Joomla** - це система управління контентом, що спочатку виникла з програми з відкритим кодом Mambo. Після розбіжностей щодо комерційного маркетингу значна частина команди розробників Mambo відвернулася і натомість зосередилася на розробці Joomla як нової CMS. Перша версія була опублікована в 2005 році, і з того часу система управління безкоштовним контентом неухильно зростає - кілька мільйонів завантажень та величезна база користувачів говорять самі за себе. Joomla, ім'я якого походить від слова суахілі *jumla* (що означає "всі разом" або "в цілому"), написано мовою сценаріїв PHP і базується на MySQL. Завдяки своїй великій популярності та дуже активному співтовариству, Joomla вже побачила тисячі доданих розширень - компонентів,

модулів, плагінів та шаблонів. Це робить навіть найскладніші проекти можливими з Joomla.

Joomla пропонує чотири типи розширень - компоненти, модулі, плагіни та шаблони. Також існує розрізнення щодо того, чи є розширення для frontend або backend. Компоненти додають такі функції, як динамічний вміст або інструменти адміністрування на сайті Joomla. Сюди входять форуми, галереї зображень, гостьові книги, карти, статистика та функції резервного копіювання. Модулі - це невеликі блоки, які дозволяють відображати дані з ядра Joomla або інших розширень у передній частині. Плагіни містять код, який можна інтегрувати в певні місця в рамках Joomla та додавати певний вміст або функціональність. Шаблони трохи відрізняються від інших розширень Joomla. Шаблони Joomla - це збірні конструкції та конструкції, які служать основою для вашого сайту і складаються щонайменше з файлів HTML і CSS. Інтегруючи один із цих шаблонів, ви зможете заощадити багато роботи при розробці веб-сайту.

### **Основні можливості**

CMS Joomla володіє безліччю можливостей, серед яких найбільш важливими є:

- Багатомовність: понад 64 доступних мов, багатомовні веб-сайти
- Прості оновлення: оновлення в один клік доступні завжди
- Управління банерами: просте розміщення оголошень, сумісне з різними системами відстеження
- Функція пошуку: інтегрований індекс пошуку, автоматично запропоновані пошукові терміни тощо.
- Просте управління вмістом: редактор WYSIWYG, безліч попередньо встановлених модулів для презентації вмісту, безліч макетів
- Редагування інтерфейсу: незначні зміни можна вносити безпосередньо в інтерфейс

- Чуйний веб-дизайн: стандартні шаблони Joomla на основі Bootstrap оптимізовані для багатьох пристроїв
- Масштабована структура MVC: окремий макет та база даних для більшої свободи в дизайні
- Вільний вибір шрифту: відсутність фіксованого шрифту за замовчуванням, проста інтеграція особистих шрифтів
- Управління користувачами: групи користувачів, управління правами тощо.
- API хмарного сховища: інтерфейси дозволяють просте підключення до хмарних сервісів, таких як Amazon S3, Google Cloud Storage або Dropbox [11].

Параметр	Значение
Платформа, на которой работает PHP	Windows NT WIN-I7UDQOP78FH 10.0 build 17763 (Windows 10) AMD64
Тип базы данных	mysql
Версия базы данных	5.5.5-10.4.16-MariaDB
Кодировка базы данных	utf8_general_ci
Кодировка соединения базы данных	utf8mb4_general_ci
Версия PHP	7.4.12
Веб-сервер	Apache/2.4.46 (Win64) OpenSSL/1.1.1h PHP/7.4.12
Интерфейс между веб-сервером и PHP	apache2handler
Версия Joomla!	Joomla! 3.9.1 Stable [ Amani ] 27-November-2018 15:00 GMT
Версия Joomla! Platform	Joomla Platform 13.1.0 Stable [ Curiosity ] 24-Apr-2013 00:00 GMT
Строка User-agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36

Рисунок 2.1.2 — Информация о CMS Joomla.

The screenshot displays the Joomla! administration interface. On the left is a navigation menu with categories: МАТЕРИАЛЫ (Create material, Manage materials, Manage categories, Media manager), СТРУКТУРА (Manage menu, Manage modules), ПОЛЬЗОВАТЕЛИ (Manage users, No pending requests), НАСТРОЙКИ (General settings, Manage templates, Manage languages), РАСШИРЕНИЯ (Extension installation), and ОБСЛУЖИВАНИЕ (Check Joomla! version, Check extension versions). The main content area features a 'Важные системные сообщения' (Important system messages) section, an 'SP PAGE BUILDER' section with 'Add New Page' and 'Pages' buttons, and three lists: 'ПОСЛЕДНИЕ 5 АВТОРИЗОВАННЫХ ПОЛЬЗОВАТЕЛЕЙ' (Last 5 authorized users), 'ПОПУЛЯРНЫЕ 5 МАТЕРИАЛОВ' (Popular 5 materials), and 'ПОСЛЕДНИЕ 5 МАТЕРИАЛОВ' (Last 5 materials).

Рисунок 2.1.3 — Меню керування.

## 2.2 Вибір технологій

Для створення Front-End частини веб-сайту, я буду використовувати такі веб-технології:

### Мова розмітки Hypertext Markup Language (HTML)

(HTML) - це комп'ютерна мова, яка полегшує створення веб-сайтів. Мова, яка має кодові слова та синтаксис, як і будь-яка інша мова, порівняно легка для сприйняття і, з часом, стає все потужнішою у тому, що вона дозволяє комусь творити. HTML продовжує розвиватися, щоб задовольнити запити та вимоги Інтернету під виглядом Консорціуму всесвітньої павутини, організації, яка розробляє та підтримує мову; наприклад, з переходом на Web 2.0.

Набір символів розмітки або кодів, вставлених у файл, призначений для відображення в Інтернеті. Розмітка підказує веб-браузерам, як відобразити слова та зображення веб-сторінки. Кожен окремий фрагмент коду розмітки (який потрапляє між символами "<" і ">") називається елементом, хоча багато людей також називають його тегом. Деякі елементи поставляються парами, які вказують, коли якийсь ефект відображення повинен початися, а коли закінчитися. [4].



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Example</title>
5     <link rel="stylesheet" href="style.css">
6   </head>
7   <body>
8     <h1>
9       <a href="/">Header</a>
10    </h1>
11    <nav>
12      <a href="one/">One</a>
13      <a href="two/">Two</a>
14      <a href="three/">Three</a>
15    </nav>
```

Рисунок 2.2.1 — Структура коду HTML

## Cascading Style Sheets (CSS)

(CSS) - означає "Каскадні таблиці стилів" з акцентом на "Стиль". Використовуються для оформлення елементів HTML, визначає стиль документа - макети сторінок, кольори та шрифти, анімації та ін. визначаються за допомогою CSS. Каскадні таблиці стилів використовуються для форматування макета веб-сторінок. Вони можуть використовуватися для визначення стилів тексту, розмірів таблиць та інших аспектів веб-сторінок, які раніше могли бути визначені лише в HTML сторінки.

CSS допомагає веб-розробникам створити єдиний вигляд кількох сторінок веб-сайту. Замість того, щоб визначати стиль кожної таблиці та кожного блоку тексту в HTML-кодї сторінки, загальноживані стилі потрібно визначати лише один раз у документі CSS. Після того, як стиль визначено в каскадній таблиці стилів, він може використовуватися будь-якою сторінкою, яка посилається на файл CSS. Крім того, CSS дозволяє легко змінювати стилі на декількох сторінках одночасно. Наприклад, веб-розробник може захотіти збільшити розмір тексту за замовчуванням з 10 до 12 пунктів для п'ятдесяти сторінок веб-сайту. Якщо всі сторінки посилаються на одну і ту ж таблицю

стилів, розмір тексту потрібно лише змінити на таблиці стилів, і всі сторінки відобразатимуть більший текст. [5].

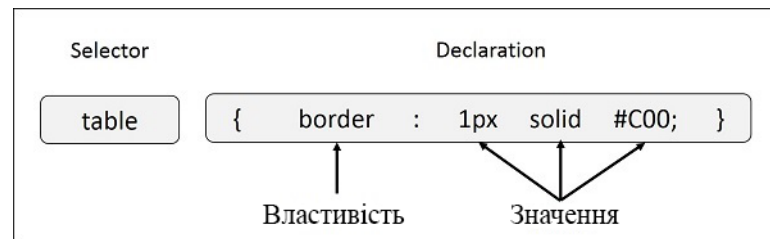


Рисунок 2.2.2 — Структура коду CSS

## Мова програмування JavaScript

(JavaScript) - це динамічна мова комп'ютерного програмування. Він легкий і найчастіше використовується як частина веб-сторінок, реалізації яких дозволяють клієнтському сценарію взаємодіяти з користувачем та створювати динамічні сторінки. Це інтерпретована мова програмування з об'єктно-орієнтованими можливостями.

JavaScript спочатку був відомий як LiveScript, але Netscape змінив свою назву на JavaScript, можливо, через хвилювання, яке створює Java. JavaScript вперше з'явився в Netscape 2.0 у 1995 році під назвою LiveScript. Ядро мови загального призначення було вбудовано в Netscape, Internet Explorer та інші веб-браузери.

Переваги JavaScript:

- JavaScript легко вивчити.
- Він виконується в браузері клієнта, тому виключає обробку на стороні сервера.
- Він виконується на будь-якій ОС.
- JavaScript можна використовувати з будь-яким типом веб-сторінки, наприклад PHP, ASP.NET, Perl тощо.
- Продуктивність веб-сторінки збільшується завдяки виконанню на стороні клієнта.

- Код JavaScript можна зменшити, щоб зменшити час завантаження із сервера.
- Багато платформ програм на основі JavaScript доступні на ринку для створення односторінкових веб-додатків, наприклад ExtJS, AngularJS, KnockoutJS тощо. [6].

## **Бібліотека Bootstrap**

(Bootstrap) – Також це називається Front-end-framework. це безкоштовна колекція інструментів для створення веб-сайтів та веб-додатків. Він містить шаблони дизайну на основі HTML та CSS для типографіки, форм, кнопок, навігації та інших компонентів інтерфейсу, а також додаткові розширення JavaScript. Шаблони CSS та JS використовуються для набору тексту, кнопок, навігації та інших взаємодіючих компонентів. Фреймворк зараз швидший і популярний для розробки мобільних веб-сайтів. Він написаний на CSS, HTML, Less, Sass та JavaScript. Bootstrap легко вивчити, а експериментування з фреймворком навряд чи потребує місяця, щоб стати екпертом у процесі. Фреймворк насправді важкий завдяки CSS та HTML файлам.

Причини чому я віддав перевагу Bootstrap Framework:

- легкість в освоєнні
- чудова система сіток
- базовий стиль для більшості елементів HTML (типографіка, код, таблиці, форми, кнопки, зображення, значки)
- великий перелік компонентів
- у комплекті плагіни Javascript [7].

## Бібліотека jQuery

(jQuery) –це бібліотека JavaScript з відкритим кодом, яка спрощує взаємодію між документом HTML / CSS, точніше об'єктною моделлю документа (DOM) та JavaScript. jQuery спрощує обхід та маніпулювання документами HTML, обробку подій браузера, анімацію DOM, взаємодію Ajax та розробку JavaScript у різних браузерах.

jQuery був розроблений Джоном Резігом та представлений у 2006 році на зборах BarCamp NYC. Джон шукав спосіб взяти загальні, повторювані блоки завдань у JS і звести всю непотрібну розмітку, щоб залишити короткі, розумні та зрозумілі «рецепти», які називаються методами. Замість того, щоб переписувати кожен блок завдань знову і знову, розробникам потрібно лише викликати їх метод - jQuery бере звідти!

jQuery займає багато загальних завдань, які вимагають виконання багатьох рядків коду JavaScript, і переносять їх у методи, які можна викликати за допомогою одного рядка коду [8].

Особливості бібліотеки jQuery:

- Селекція DOM: jQuery надає селектори для отримання елемента DOM на основі різних критеріїв, таких як ім'я тегу, ідентифікатор, назва класу css, ім'я атрибута, значення, дочірня частина в ієрархії тощо.
- Маніпуляція DOM: Ви можете маніпулювати елементами DOM, використовуючи різні вбудовані функції jQuery. Наприклад, додавання або видалення елементів, зміна вмісту html, клас css тощо.
- Спеціальні ефекти: Ви можете застосувати спеціальні ефекти до елементів DOM, наприклад, показати або приховати елементи, вбудовування або зменшення видимості, ефект ковзання, анімацію тощо.

- Події: бібліотека jQuery включає функції, еквівалентні подіям DOM, такі як `click`, `dblclick`, введення миші, залишення миші, розмиття, клавіатура, клавіатура тощо. Ці функції автоматично обробляють проблеми між браузерами.
- Ajax: бібліотека jQuery включає також прості у використанні функції AJAX для завантаження даних із серверу без перезавантаження веб-сторінки.
- Підтримка браузерів: бібліотека jQuery автоматично обробляє проблеми між браузерами, тому користувачі не доведеться про це турбуватися. Також бібліотека jQuery підтримує IE 6.0+, FF 2.0+, Safari 3.0+, Chrome та Opera 9.0+.

### **Мова програмування PHP**

(PHP) – це мова сценаріїв OpenSource, яка зазвичай використовується для веб-розробки. Він використовує сценарії на стороні сервера, за допомогою яких скрипт встановлюється на сервері, щоб сервер міг надавати результати, що залежать від введення кінцевого користувача, особливо популярний для веб-розробки, оскільки його легко вбудувати в HTML-сторінки.

Сам PHP використовує початковий і кінцевий коди, так що веб-розробнику легко перейти на PHP і вийти з нього.

Однією з головних переваг PHP є те, що він має легку криву навчання, щоб новачки могли швидко впоратися з основами. Але PHP також надзвичайно потужний.

Окрім виконання простих завдань, таких як збір даних користувачів, введення за допомогою онлайн-форми, та створення динамічних сторінок, PHP також має більш розширені функції. Зазвичай він використовується для сценаріїв на стороні сервера, сценаріїв командного рядка і навіть для створення настільних додатків, та сумісний

з усіма основними операційними системами, включаючи Linux, Windows та MAC OS. Він також сумісний з більшістю популярних веб-серверів, таких як Apache та LiteSpeed.

PHP може використовуватися для створення динамічного вмісту, такого як PDF-файли, зображення та відео, і сумісний з найпопулярнішими типами інструментів баз даних, такими як MySQL. Для створення деяких найпопулярніших програм, таких як інструменти ведення блогу, такі як WordPress та Joomla, а також кошики для електронної комерції, такі як OpenCart та Prestashop, а також інструменти дошки оголошень, такі як PHPBB та vBulletin.

Також можна використовувати для додавання додаткових функціональних можливостей до вашого веб-сайту, інтегруючи додатки, створені для сторонніх веб-сайтів, таких як Facebook, або створюючи списки розсилки на основі PHP, або ви навіть можете створювати плагіни WordPress для комерційного використання.

Переваги:

- **Відкритий код** – його розробляє і підтримує велика група розробників. Це допоможе у створенні спільноти підтримки та великих бібліотек розширень.
- **Швидкість** – Це відносно швидко, оскільки використовує не так багато системних ресурсів.
- **Простий у використанні** – він використовує синтаксис, подібний до C, тому для тих, хто знайомий з C, їм дуже легко підібрати та легко створити сценарії веб-сайтів.
- **Стабільність** – оскільки його підтримують багато розробників, помилки досить швидко знаходять і виправляють, роблячи його стабільним програмним забезпеченням.
- **Потужна підтримка бібліотеки** – Ви можете легко знайти необхідні вам функціональні модулі, такі як PDF, графік тощо.

- **Вбудовані модулі підключення до бази даних** – Ви можете легко підключатися до баз даних за допомогою PHP, оскільки багато веб-сайтів керуються даними / вмістом, тому ми будемо часто використовувати базу даних, це значною мірою скоротить час розробки веб-програм. [9].

## База даних MySQL

MySQL - це вільно доступна реляційна система управління базами даних (СУБД), яка використовує структуровану мову запитів (SQL).

SQL - найпопулярніша мова для додавання, доступу та управління вмістом у базі даних. Найбільш відомий своєю швидкою обробкою, доведеною надійністю, простотою та гнучкістю використання. MySQL є важливою частиною майже кожного PHP-додатку з відкритим кодом. Хорошими прикладами для сценаріїв на основі PHP та MySQL є WordPress, Joomla! Та Drupal.

Переваги бази даних MySQL:

- MySQL може підтримувати різні механізми баз даних, тоді як інші типи систем не підтримують таку функцію.
- У порівнянні з іншими реляційними базами MySQL характеризується високою продуктивністю.
- Економічно MySQL є досить ефективною базою. Це набагато дешевше, ніж деякі інші бази реляційного типу. Крім того, ціна ліцензії на комерційну версію також не така висока.
- MySQL - це міжплатформова база.

Раніше типом зберігання даних у MySQL за замовчуванням був MyISAM, який був дуже простий у використанні та примітний високою продуктивністю. На сьогодні він використовується InnoDB (версія 5.5), який підтримує зовнішні ключі для цілісності посилань. [10].

## 2.3 Вибір структури патерну

Модель-View-Controller (MVC) - це архітектурний шаблон, який розділяє додаток на три основні логічні компоненти: модель, вигляд та контролер. Це робиться для відокремлення внутрішнього подання інформації від способів подання та прийняття інформації від користувача. Традиційно використовуваний для настільних графічних програм, шаблон став популярним з появою веб-програм. Сьогодні майже всі популярні мови підтримують цю архітектуру.

Компоненти:

- Модель є центральним компонентом патерна. Динамічна структура даних програми, яка незалежна від інтерфейсу користувача. Це відповідає всій логіці, пов'язаній із даними, з якою працює користувач. Він безпосередньо управляє даними, логікою та правилами програми.
- Видом може бути будь-яке вихідне подання інформації, наприклад діаграма або діаграма. Можливі декілька подань однієї і тієї ж інформації, наприклад, гістограма для управління та таблична таблиця для бухгалтерів.
- Третя частина або розділ, контролер, приймає введення та перетворює їх на команди для моделі чи подання. Він діє як інтерфейс між компонентами Model і View для обробки всієї бізнес-логіки та вхідних запитів, маніпулює даними за допомогою компонента Model і взаємодіє з поданнями, щоб зробити кінцевий результат [12].



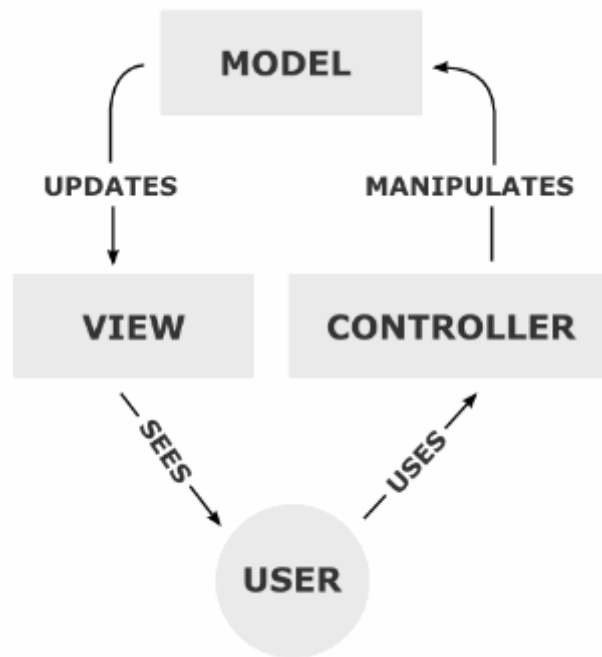


Рисунок 2.3.1 — Структура Model-View-Controller

## 2.4 Вибір та установка локального серверу

Локальний сервер - спеціальна програма, яка дозволяє веб-розробникам розробляти сайти на локальному комп'ютері без доступу до Інтернету.

Велика частина сайтів сучасного інтернету динамічні і працюють в основному на PHP. PHP також часто використовується для навчання програмуванню. Але браузері розуміють тільки HTML і CSS, а PHP - немає. Тому що PHP - це серверний мова програмування і сервер якраз перетворює і обробляє PHP-код (або результат його виконання) в вид, зрозумілий браузеру.

І такі обробники стоять на кожному сервері / хостингу в інтернеті але не на вашому домашньому комп'ютері.

Потреба в локальних серверах постійно росла при розробці динамічних сайтів на PHP, Перлі та іншими мовами програмування. Спочатку це було обумовлено поганим і дорогим інтернетом, потім люди зрозуміли необхідність

тестування скриптів в спеціальному середовищі, та й взагалі виріс аматорський і професійний інтерес до програмування.

Для створення власного локального серверу я обрав XAMPP - це дистрибутив програмного забезпечення, який забезпечує веб-сервер Apache, базу даних MySQL (насправді MariaDB), Php та Perl (як виконувані файли командного рядка та модулі Apache) в одному пакеті. Він доступний для систем Windows, MAC та Linux. Для інтеграції Php з MySQL не потрібна конфігурація. Важливою частиною XAMPP є Apache, який використовується для запуску веб-сервера на вікнах. Запускаючи цей локальний сервер у вікнах, веб-розробник може протестувати свої веб-сторінки. За допомогою цього локального сервера можна перевіряти динамічні веб-сайти, не публікуючи їх на живому веб-сервері. Це простий у використанні і код з PHP. Він доступний як для 32-бітової, так і для 64-бітової системи [12].

Особливості:

- Керування службами Apache, MySQL і MariaDB;
- Встановлюйте та перемикайте версії Apache, MySQL, MariaDB та PHP;
- Керування налаштуваннями серверів;
- Отримання доступу до своїх журналів;
- Доступ до файлів налаштувань.

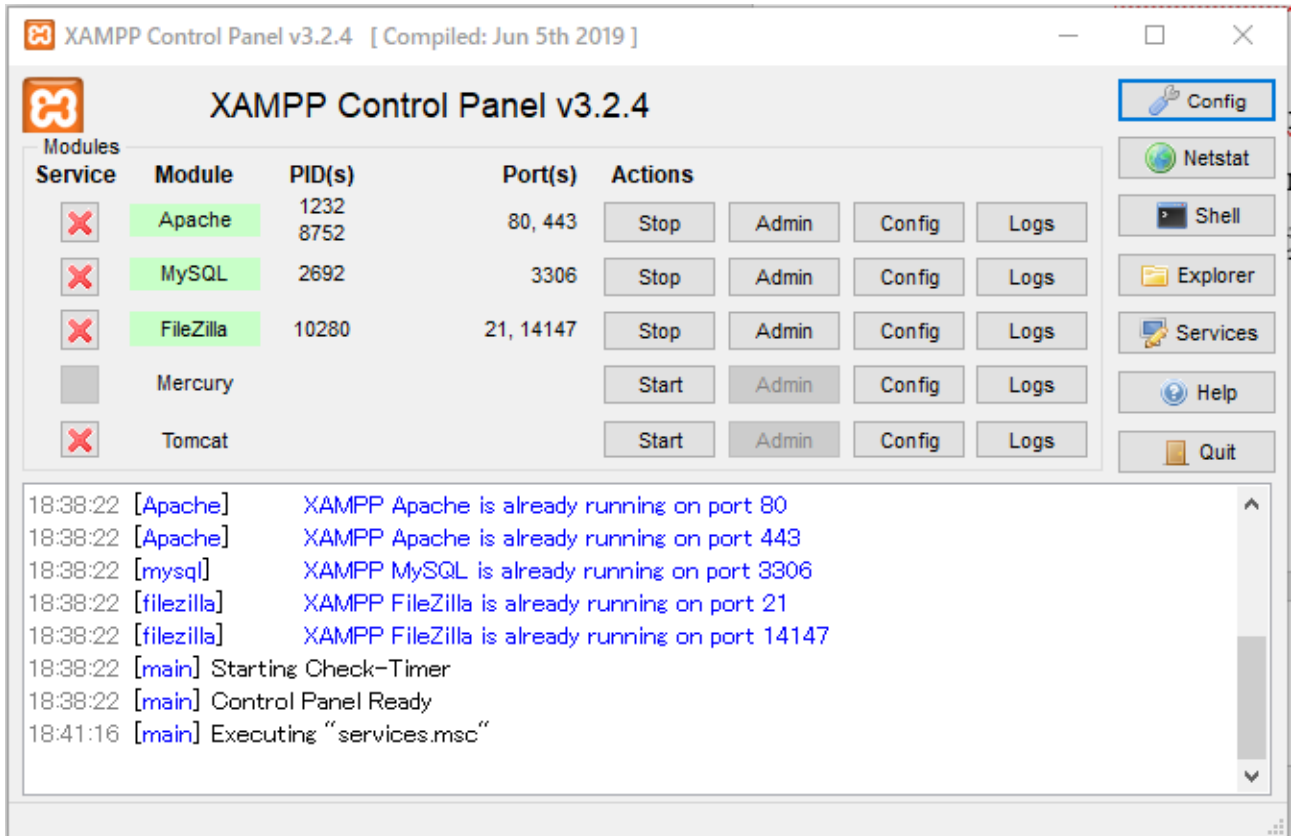


Рисунок 2.4.1 — Меню керування XAMPP

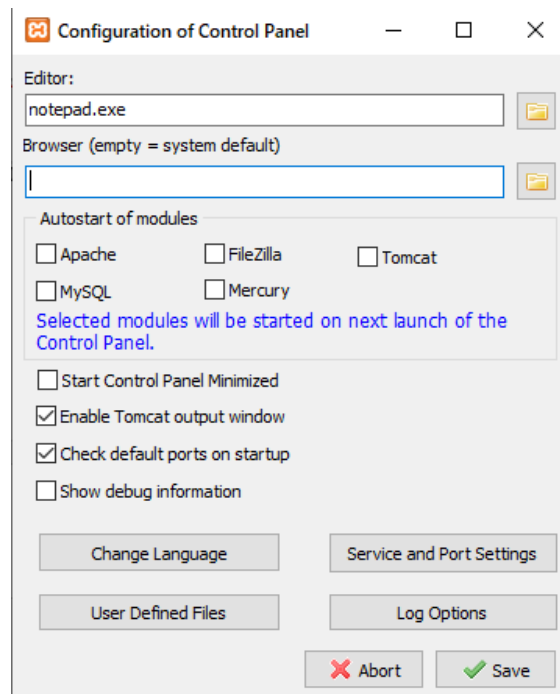


Рисунок 2.4.2 — Налаштування XAMPP

## 2.5 Вибір редактора коду

В якості редактора коду був обраний Sublime Text, він доступний для OS X, Windows та Linux. Для використання Sublime Text на кожному вашому комп'ютері, незалежно від того, яку операційну систему він використовує, вам потрібна одна ліцензія. Sublime Text використовує спеціальний набір інтерфейсів користувача, оптимізований для швидкості та краси, одночасно користуючись перевагами власних функцій на кожній платформі.

Sublime Text - це інструмент у категорії Текстовий редактор технічного стеку. Серед основних причин, чому я почав використовувати Sublime Text замість інших програм так як:

- За допомогою різноманітних плагінів програма здається, що вона побудована навколо розробника та спроектована для зменшення натискань клавіш та використання миші.
- У Sublime Text є сотні корисних шорткатів для команд, яких не існує в Dreamweaver. Ярлики для таких речей, як загортання тегів, дублювання рядків та вибір дужок, щоб назвати лише декілька.
- Просунутий пошук, означає, що вам потрібно ввести лише декілька ключових символів, і Sublime Text спробує зіставити їх зі списком найбільш важливих команд, на його думку. Наприклад, на палітрі команд, ввівши „IP”, з'явиться команда „Встановити пакет”. У CSS, набравши "bco", з'явиться декларація "background-color".
- Однією з можливостей, яка може бути дуже корисною, є стислий попередній перегляд макета коду, особливо для великих сторінок. Це дозволяє швидко переглядати та вибирати області коду на основі загальної форми та структури.
- Sublime Text має величезну і активну спільноту плагінів, і з тисячами доступних плагінів будь-які функції, які ви можете пропустити від інших текстових редакторів, ймовірно, існуватимуть у сховищі плагінів Sublime.

- Також ви можете встановити власні теми, наприклад, для зміни графічного інтерфейсу та кольорів коду. Плагіни та прив'язки ключів дозволяють покращити процес кодування, а вікно редактора коду розділення дозволяє працювати над різними файлами поруч.

```

1 <?php
2 /**
3  * @package   Jr_tfb
4  * @subpackage com_jr_tfb
5  * @version   1.0
6  * @license   GNU/GPL
7  * @author    MeA.L.
8  * @author    BloodySonic@gmail.com
9  * @website   http://jrextensions.ru
10  */
11
12 header("Content-type: text/html; charset=utf-8");
13 define('JEXEC', 1);
14 define('DS', DIRECTORY_SEPARATOR);
15 define('JPATH_BASE', realpath(dirname(__FILE__) . '/../..../' ));
16 require_once(JPATH_BASE.DS.'includes'.DS.'defines.php');
17 require_once(JPATH_BASE.DS.'includes'.DS.'framework.php');
18 //require_once(JPATH_BASE.DS.'libraries'.DS.'joomla'.DS.'factory.php');
19 $app = JFactory::getApplication('site');
20 $app->initialise();
21 // Connect DB
22 $db = &JFactory::getDBO();
23 // Get params
24 jimport('joomla.application.component.helper' );
25 jimport('joomla.html.parameter');
26 $params = &JComponentHelper::getParams('com_jr_tfb');
27 // Connect language
28 $lang = JFactory::getLanguage();
29 $extension = 'com_jr_tfb';
30 $base_dir = JPATH_SITE;
31 if ($_POST['language'] != '') $language_tag = $_POST['language'];
32 else $language_tag = 'ru-RU';
33 $reload = true;
34 $lang->load($extension, $base_dir, $language_tag, $reload);
35
36 if(isset($_SERVER['HTTP_X_REQUESTED_WITH']) && !empty($_SERVER['HTTP_X_REQUESTED_WITH']) && strtolower($_SERVER['HTTP_X_REQUESTED_WITH']) == 'xmlhttprequest') {
37
38
39
40 // Get data
41 $test_id = $_POST['jr_tfb_id'];
42 $current_user_id = $_POST['jr_tfb_user'];
43
44
45

```

Рисунок 2.5.1 — Головне вікно програми Sublime

### 3. Проектування та реалізація веб-ресурса “Studmeplease”

#### 3.1 Реалізація інтерфейсу веб-ресурса користувача

В структурі сайту буде головна сторінка завдяки цієї сторінки користувач зможе переходити в інші меню щоб проходити курси, теорію і тд.

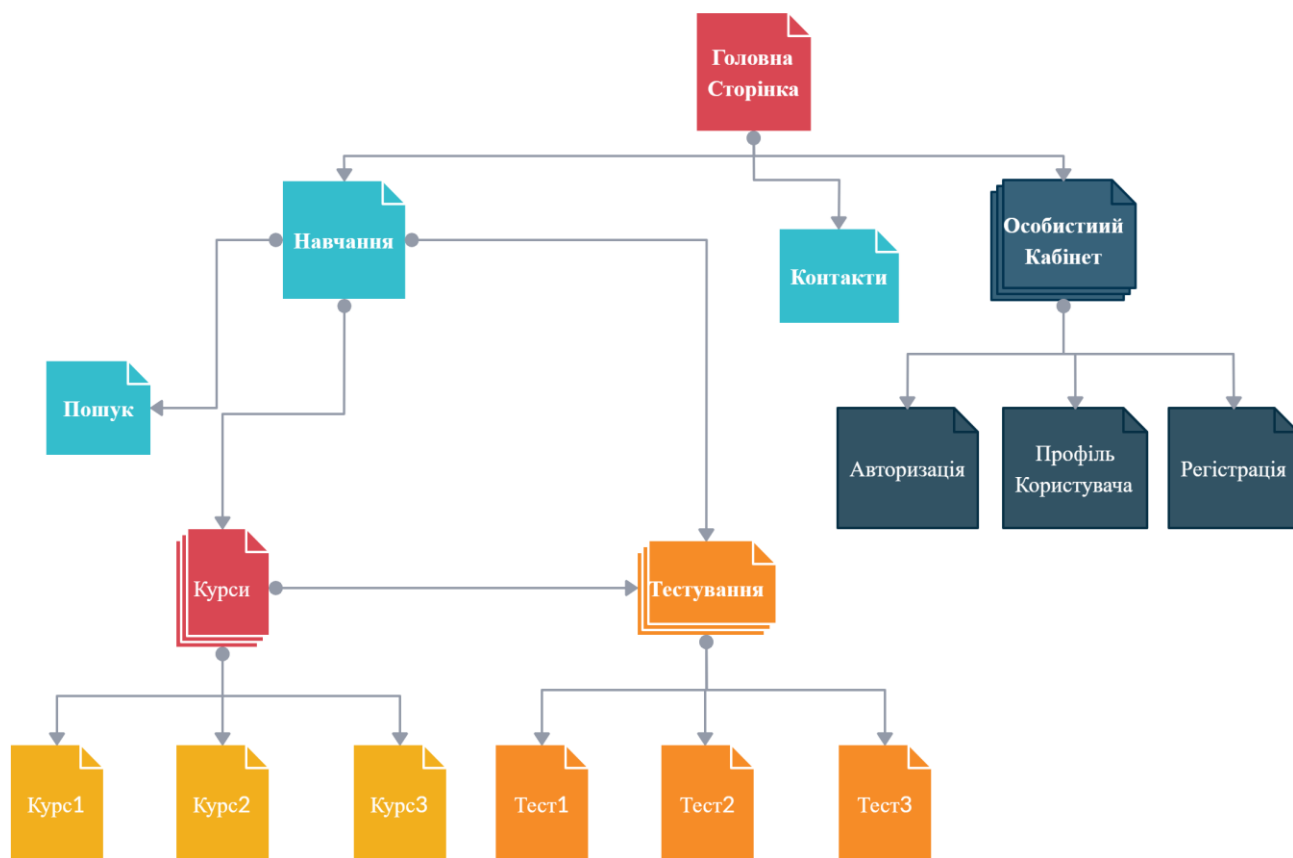


Рисунок 3.1.1 — Діаграма структури веб-ресурса

Згідно з розробленою структурою була спроектована головна сторінка Web-сайту. До її складу входять всі основні структурні елементи, перехід за якими здійснюється за допомогою гіперпосилань.

У розробку сайту входить процес створення макета Web-сторінок, на який згодом будуть нанизуватися всі інші елементи. При цьому відбувається формування так званих структурних блоків сайту - відокремлених модулів, кожен з яких відіграє певну роль і відповідає за певний функціонал ресурсу.

Склад Web-сайту включає в себе:

#### 1. Шапка сайту.

Самий верхній блок часто називають шапкою сайту, або або хедер від англ. header. Місце, в якому зазвичай розташовується хедер - верхня частина сторінки. Тут розташовуються ключові елементи навігації. Виглядає це так(рисунок 3.1.2).

В даному блоці розміщується:

- Назва сайту(лого)
- Головне меню
- Модулі реєстрації та авторизації
- Контакти
- Навчання
- Меню гамбургер
- Модуль пошуку
- Швидкі переходи до інших зв'язаних ресурсів

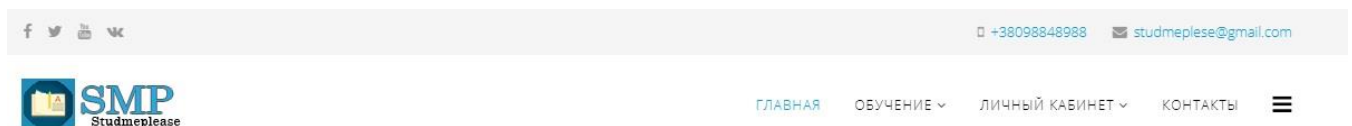


Рисунок 3.1.2 — Хедер розроблюваного веб-ресурса

## Приклад таблиці стилів шаблону хедера.

### Головне меню:

```

al-header {
    background: #ffffff none repeat scroll 0 0;
    box-shadow: 0 3px 3px rgba(0,0,0,0.05);
    height: 90px;
    left: 0;
    position: relative;
    top: 0;
    width: 100%;
    z-index: 99;
}

.al-megamenu-parent .al-dropdown li.al-menu-item {
    display: block;
    padding: 0;
    position: relative;
}

.al-megamenu-parent .al-dropdown li.al-menu-item >a,
.al-megamenu-parent .al-dropdown li.al-menu-item a {
    display: block;
    padding: 10px;
    color: #141414;
    cursor: pointer;
}

.al-megamenu-parent .al-dropdown li.al-menu-item >a.al-group-title,
.al-megamenu-parent .al-dropdown li.al-menu-item a.al-group-title {
    text-transform: uppercase;
    font-weight: bold;
}

```

### Лого:

```

}
#al-header .logo {
    display: inline-block;
    height: 90px;
    display: table;
    -webkit-transition: all 400ms ease;
    transition: all 400ms ease;
    margin: 0;
}
#al-header .logo a {
    font-size: 22px;
    line-height: 1;
}

```



```

margin: 0;
padding: 0;
vertical-align: middle;
display: table-cell;

}
#al-header .logo p {
margin: 6px 0 0;
}
#al-header .al-retina-logo {
display: none;

```

## 2. Основна область сайту.

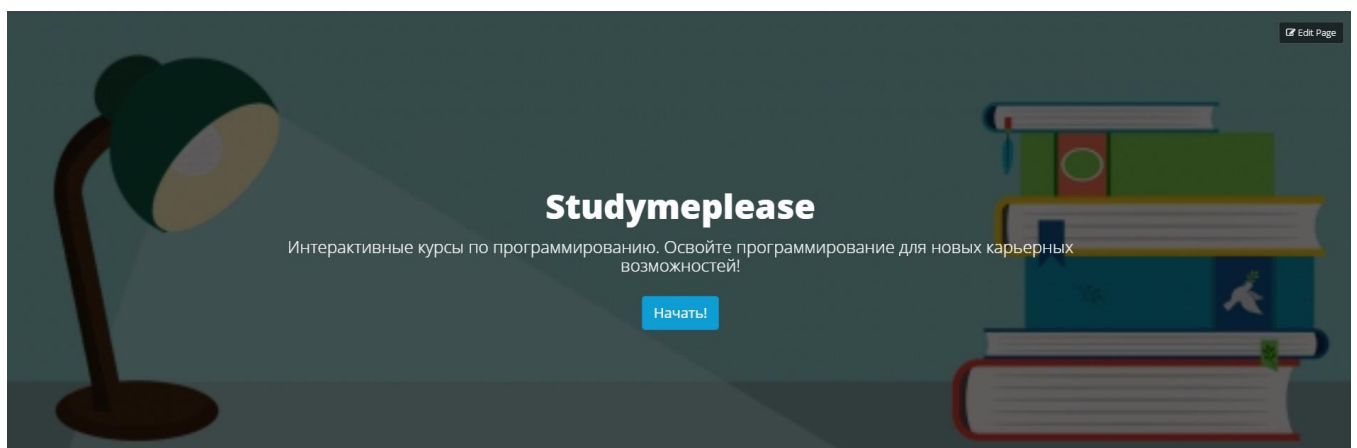
Слово «Контент» походить від англійського «content» - зміст. Це найбільша за площею і найбільш важлива для відвідувачів сайту частина сторінки.

В даному блоці розміщується:

- текстовий контент;
- графічний контент.

Ширина області основного контенту може варіюватися в досить широких межах - тут все залежить від типу макета сайту (жорсткий або гумовий). Гумовий макет дозволяє змінювати ширину блоку в залежності від дозволу монітора, з якого користувач переглядає сайт. Жорсткий макет цього зробити не дозволяє.

За допомогою функцій шаблону в Joomla був створений слайдер з кнопкою який відображає головний мотив сайту сайту. (рисунок 3.1.3).



### Рисунок 3.1.3 — Слайдер с інформацією та кнопкою з прямим переходом до матеріалів.

#### Приклад слайдера:

```
.alm-addon-cta .alm-cta-subtitle {
  margin-bottom: 0;
  line-height: 26px !important;
}
.alm-addon-cta .alm-btn-lg {
  margin-top: 30px;
}
.alm-addon-image-content {
  position: relative;
}
.alm-addon-image-content .alm-image-holder {
  top: 0;
  width: 51%;
  height: 100%;
  background-size: cover;
  background-position: 51%;
  position: absolute;
}
.alm-addon-image-content.almigment-left .alm-image-holder {
  left: 0;
}
.alm-addon-image-content.almigment-left .alm-content-holder {
  padding: 125px 0 125px 45px;
}
.alm-addon-image-content.almigment-right .alm-image-holder {
  right: 0;
}
.alm-addon-image-content.almigment-right .alm-content-holder {
  padding: 120px 50px 120px 0;
}
@media (max-width: 768px) {
  .alm-addon-image-content .alm-image-holder {
    width: 100% !important;
    height: 290px;
    position: inherit;
  }
}
```

Далі йдуть блоки інфо які створені за допомогою бібліотеки bootstrap в шаблоні де користувач може переглянути переваги та опис веб-ресурсу (рисунки 3.1.4 – 3.1.5).

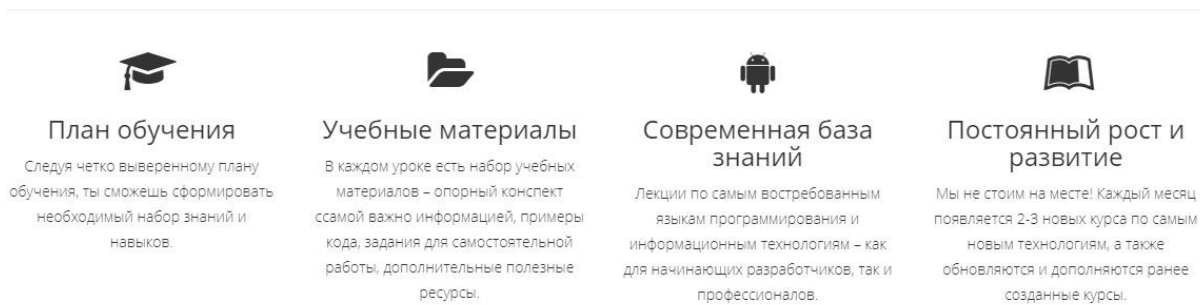


Рисунок 3.1.4 — Блок инфо переваг веб-ресурсу.

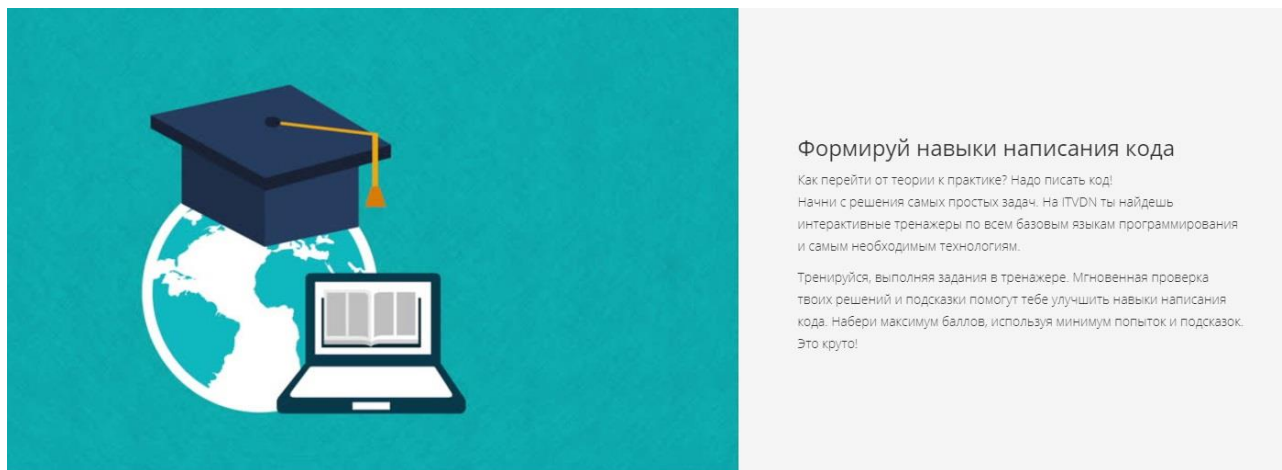


Рисунок 3.1.5 — Контейнер з описом веб-ресурсу.

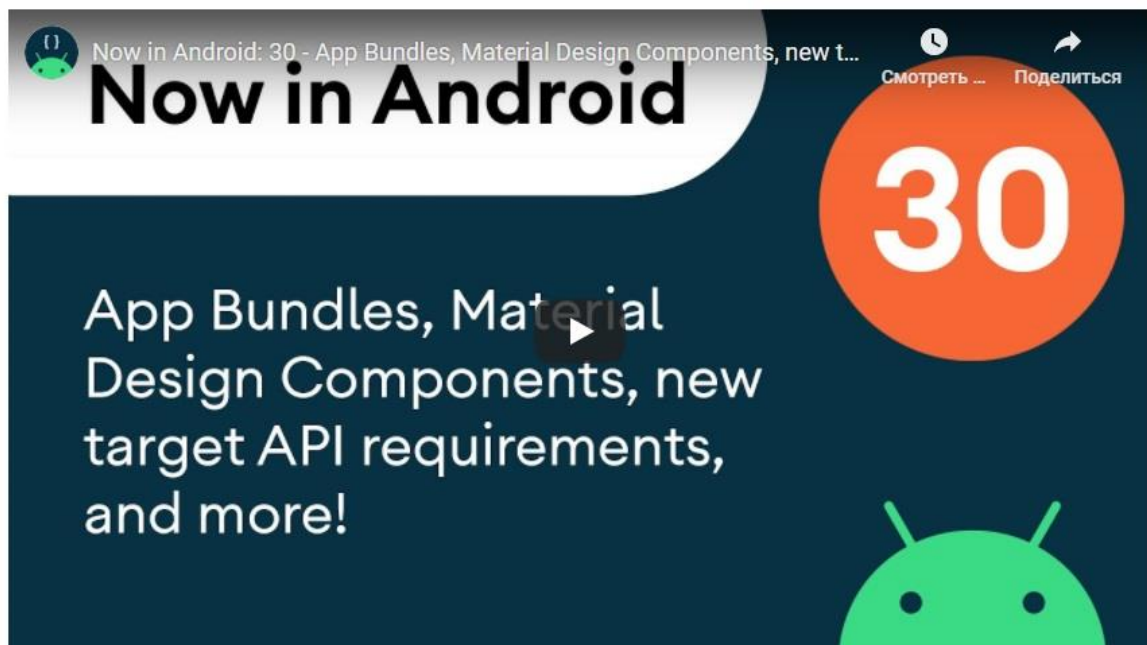
Користувач також може переглянути вступ та опис прикріпленого курсу. Він був зроблений за допомогою компоненти для Joomla “Simple Portfilio”. (рисунки 3.1.6 – 3.1.7).

## Наши курсы

Чему ты хочешь научиться? Какие программы создавать? Веб-сайты, бизнес-приложения, игры, сервисные приложения для мобильных устройств? Или тебе интересна работа с данными? Прислушайся к себе. Тебе помогут записи вебинаров, в которых рассмотрены особенности разных профессий и путь новичка. Следуй советам тех, кто добился успеха и ты достигнешь цели.



Рисунок 3.1.6 — Модуль “Simple Portfolio” на головной странице веб-ресурса



## Android Developer

### Краткое описание

Во второй половине 2016 года общая доля смартфонов на ОС Android, проданных клиентам, составила 87%, обогнав не менее популярную платформу iOS. Спрос растет, а вот количество толковых разработчиков для Android не увеличивается, что связано с минимальной популярностью профессии в странах СНГ.

Создание программных продуктов для девайсов — увлекательный и очень прибыльный процесс, ведь сегодня каждая компания, занимающаяся бизнесом, стремится выпустить собственное приложение для взаимодействия с клиентами.

### Date

09 декабря 2020

### Tags

Обучение

Рисунок 3.1.7 — Детальный обзор курсу

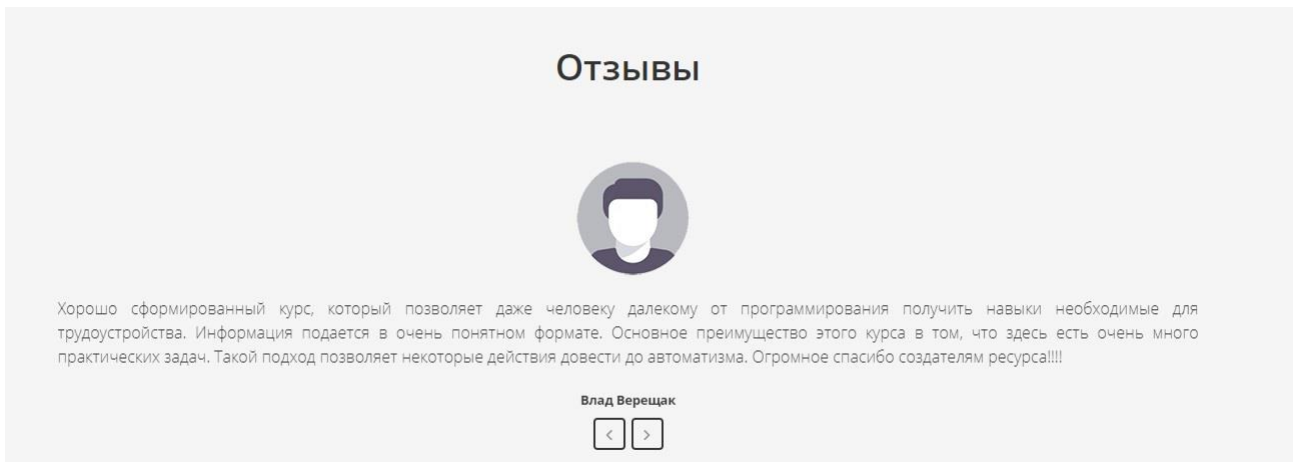


Рисунок 3.1.8 — Слайдер відгуків

### 3. Основна область сайту.

Поняття «Сайдбар» походить від англійського терміна "sidebar", де "side" - сторона, "bar" - смуга. Зазвичай сайдбарі називають бічну колонку сайту, яка розміщується праворуч або ліворуч від області основного контенту. У деяких випадках на сайті можуть розташовуватися відразу два сайдбара (перший - зліва від основної області, а другий - праворуч).

Як правило, вміст сайдбара не змінюється від однієї сторінки сайту до іншого, на відміну від вмісту області основного контенту. Тому в сайдбарі зазвичай розміщують блоки з посиланнями, а також важливу службову інформацію. У сайдбар розроблюваного сайту входить:

- Реклама інших ресурсів;
- новинний модуль показу останніх матеріалів.
- модуль пошуку

Ширина сайдбара, як правило, чітко фіксована і не залежить від типу сайту (рисунок 3.1.9):

## ПОИСК

## ПОСЛЕДНИЕ МАТЕРИАЛЫ

Лекция 1. Назначение и применение  
JavaScript, общие сведения

09 декабря 2020

---

Лекция 2: Типы данных и операторы

09 декабря 2020

---

Лекция 3: Функции и объекты

09 декабря 2020

---

Лекция 1. Введение в  
программирование на языке Python

09 декабря 2020

---

Лекция 2: Основные стандартные  
модули Python

09 декабря 2020

---

Лекция 3. Элементы  
функционального  
программирования

09 декабря 2020

## УЗНАЙТЕ БОЛЬШЕ!

> УспехЮА

---

> ITEA

---

> ВсеКурсы

---

> Skillchek

Рисунок 3.1.9 — Сайдбар веб-ресурсу

#### 4. Футер веб-сайту (подвал).

Слово «Футер» походить від англ. «Footer» - підвал. Зазвичай футером називають частини сайту, яка розташована під усіма іншими блоками.

За аналогією з шапкою сайту, футер також має альбомну орієнтацію.

Ширина футера також може змінюватися в залежності від дозволу монітора користувача (рисунок 3.1.10).

У підвалі сайту розміщені:

- дублікат інформації про сайт;
- друзі/партнери сайту;
- дублікат основного меню;
- останні лекції та тести

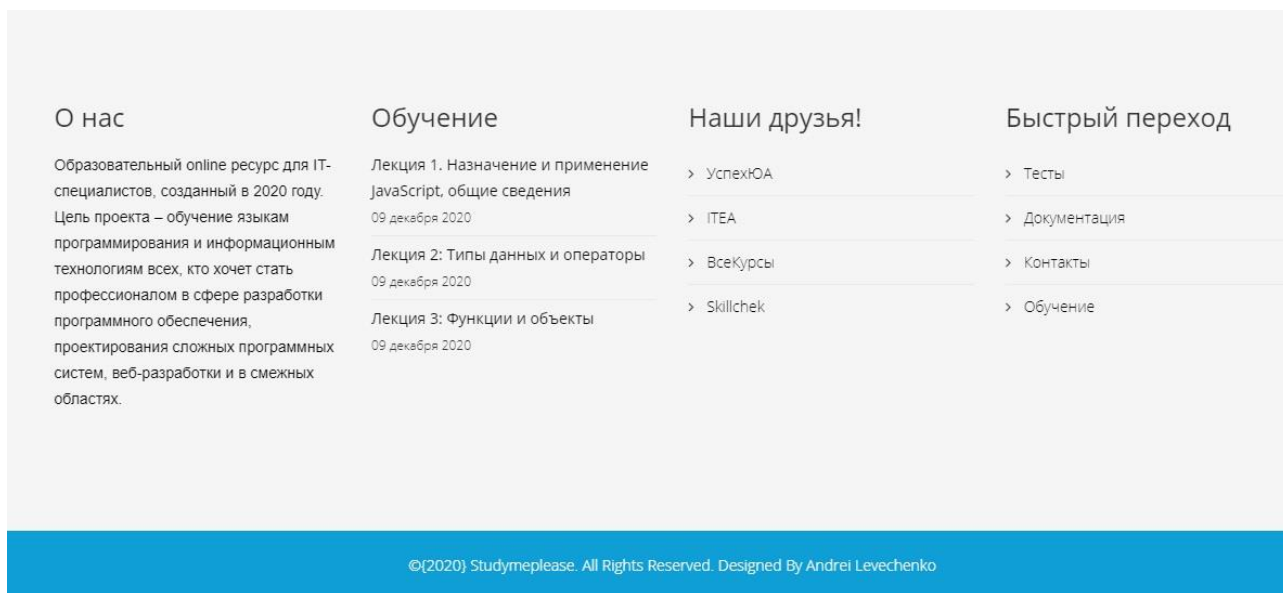


Рисунок 3.1.10 — Футер веб-ресурсу

Для проходження тестування на курсу, користувачу необхідно буде зареєструватися в веб-додатку. (Рисунки 3.1.11-3.1.12)

Имя \*

Логин \*

Пароль \*

Подтвердите пароль \*

Адрес e-mail \*

Подтвердите e-mail \*

[Регистрация](#) [Отмена](#)

Рисунок 3.1.11 — Модуль форми реєстрації

Bloodysonic

.....

Запомнить меня

[Войти](#)

[Забыли пароль?](#)

[Забыли логин?](#)

[Еще нет учетной записи?](#)

Рисунок 3.1.12 — Модуль форми авторизації



На рисунку 3.1.13 зображений зворотній зв'язок за допомогою якого користувач може зв'язатися з адміністрацією веб-ресурса.

Форма обратной связи

Отправка сообщения

\* Обязательное поле

Имя \*

E-mail \*

Тема \*

Сообщение \*

Отправить копию   
сообщения на ваш  
адрес  
(необязательно)

Рисунок 3.1.13 — Форма зворотнього зв'язку

На рисунку 3.1.14 можна побачити сторінку с лекцією по предмету Java.



The image shows a screenshot of a video lecture page. At the top, there is a video player with a red background and the Java logo (a yellow coffee cup with steam) and the word 'Java' in white. Below the video player, there is a blue icon of a video camera and the text 'ЯЗЫК ПРОГРАММИРОВАНИЯ JAVA' (Java Programming Language), '09 ДЕКАБРЯ 2020' (December 9, 2020), and 'ПРОСМОТРОВ: 78' (Views: 78). The main title of the lecture is 'Лекция 1: Что такое Java? История создания' (Lecture 1: What is Java? History of creation). Below the title, there is a section 'Страница 1 из 9' (Page 1 of 9) and 'Что такое Java?' (What is Java?). The text under this heading asks: 'Что знают о Java обычные пользователи персональных компьютеров и Internet? Что говорят о нем разработчики, которые не занимаются этой технологией профессионально?' (What do ordinary users of personal computers and Internet know about Java? What do developers who do not work professionally with this technology say about it?). Below this, there is a paragraph about Java: 'Java широко известна как новейший объектно-ориентированный язык, легкий в изучении и позволяющий создавать программы, которые могут исполняться на любой платформе без каких-либо доработок (кроссплатформенность). Еще с Java почему-то всегда связана тема кофе (изображения логотипов, названия продуктов и т.д.). Программисты могут добавить к этому описанию, что язык похож' (Java is widely known as the latest object-oriented language, easy to learn and allowing to create programs that can be executed on any platform without any modifications (cross-platform). Still with Java, for some reason, the topic of coffee is always associated (logos, product names, etc.). Programmers can add to this description that the language is similar).

Содержание материала (Content of the material) section includes a list of topics: 'Лекция 1: Что такое Java? История создания' (Lecture 1: What is Java? History of creation), 'История создания Java' (History of creation of Java), 'Word Wide Web', 'История развития Java' (History of development of Java), 'Сетевые компьютеры' (Network computers), and 'Платформа Java' (Java platform).

Рисунок 3.1.14 — Детальный обзор лекції

Після лекції користувач повинен пройти тестування. Рисунки 3.1.15-3.1.16

Время тестирования: 00:00:10

### Лекция 1: Java тест

Что будет выведено на консоль при вычислении следующего выражения?

```
public class Test{
    public static void main(String[] args){
        char c1 = '1';
        char c2 = '\u0031';
        char c3 = 49;
        System.out.println(c1 + c2 + c3);
    }
}
```

147

выберите вариант ответа

111

Ошибка компиляции

**147**

81

Рисунок 3.1.15 — Початок тестування

### Тест пройден

Набрано баллов: 6

Ответы на вопросы

Что будет выведено на консоль при вычислении следующего выражения? 147

Какие из следующих строк скомпилируются без ошибок?

```
int[ ][ ] scores = {{1, 2}, {1, 2, 3, 4}, {1, 2, 3}};
String[ ] names[ ];
float[ ] f1[ ], f2;
```

Что будет результатом выполнения программы? Ничего из вышперечисленного

Что будет выведено на консоль при вычислении следующего выражения? true

Что произойдет при попытке скомпилировать и запустить данный код? Считается, что файл ostream уже создан. Ошибка компиляции

Что напечатает следующий фрагмент кода: finally

Рисунок 3.1.16 — Закінчення тестування

### 3.2 Опис інтерфейсу адміністратора

Вигляд інтерфейсу з боку адміністратора. Меню, матеріали, компоненти та модулі.

<input type="checkbox"/>	<input checked="" type="checkbox"/>	Главная (Алиас: home) SP Page Builder » Page	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Обучение (Алиас: blog) Материалы » Блог категории	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	– Курсы (Алиас: blog-with-right-sidebar) – Материалы » Список всех категорий	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	– Поиск (Алиас: search) – Поиск » Форма поиска и список результатов поиска	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	– Тестирование (Алиас: testi) – Тестирование » Список категорий	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Личный кабинет Ссылка (URL)	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	– Авторизация (Алиас: login-form) – Пользователи » Форма авторизации	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	– Профиль пользователя (Алиас: user-profile) – Пользователи » Профиль пользователя	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	– Изменить (Алиас: edit-user-profile) – Пользователи » Изменить профиль	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	– Регистрация (Алиас: registration-form) – Пользователи » Форма регистрации	Мейн Меню
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Контакты (Алиас: kontakty) Контакты » Контакт	Мейн Меню

Рисунок 3.2.1 — Пункты меню в адмін панелі

The screenshot shows the admin interface for 'SP Simple Portfolio: Items'. At the top, there are buttons for 'Создать', 'Редактировать', and 'В корзину'. Below these is a search bar and a sidebar with 'Items', 'Categories', and 'Tags'. The main content area displays a list of items with columns for 'Image', 'Заголовок', and 'Доступ'. The items listed are:

Image	Заголовок	Доступ
	Android Developer (Алиас: corporate-template) Tags: Обучение	Public
	Java developer (Алиас: helix-free-template) Tags: Обучение	Public
	Python Developer (Алиас: free-joomla-template) Tags: Обучение	Public
	Frontend Developer (Алиас: responsive-news-template) Tags: Magazine	Public

Рисунок 3.2.2 — Вигляд модуля “Simple Portfolio” в адмін панелі

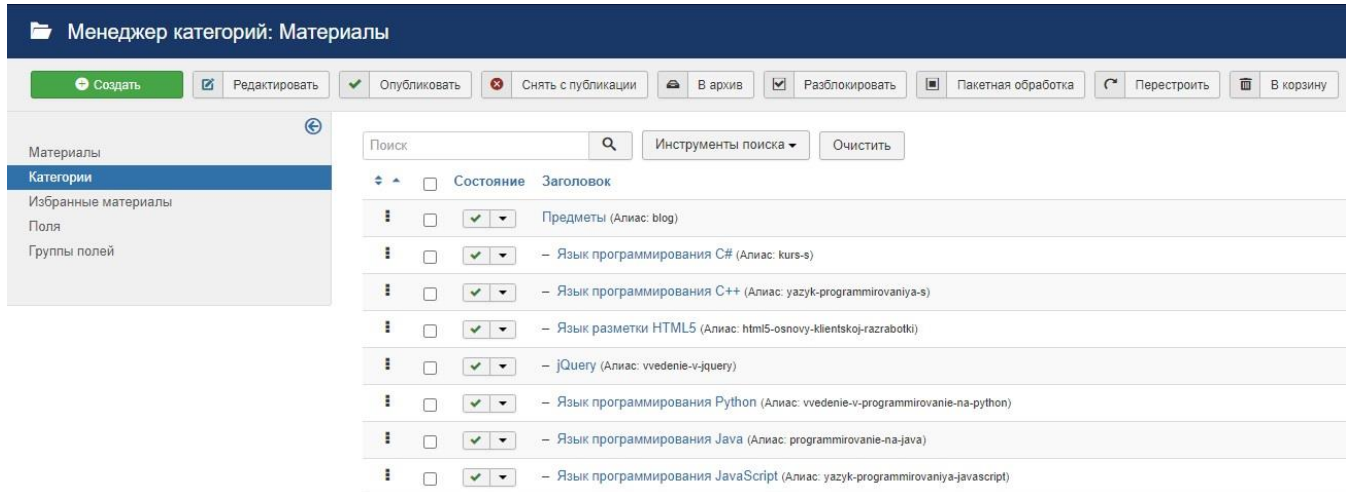


Рисунок 3.2.3 — Початковий набір категорій курсів

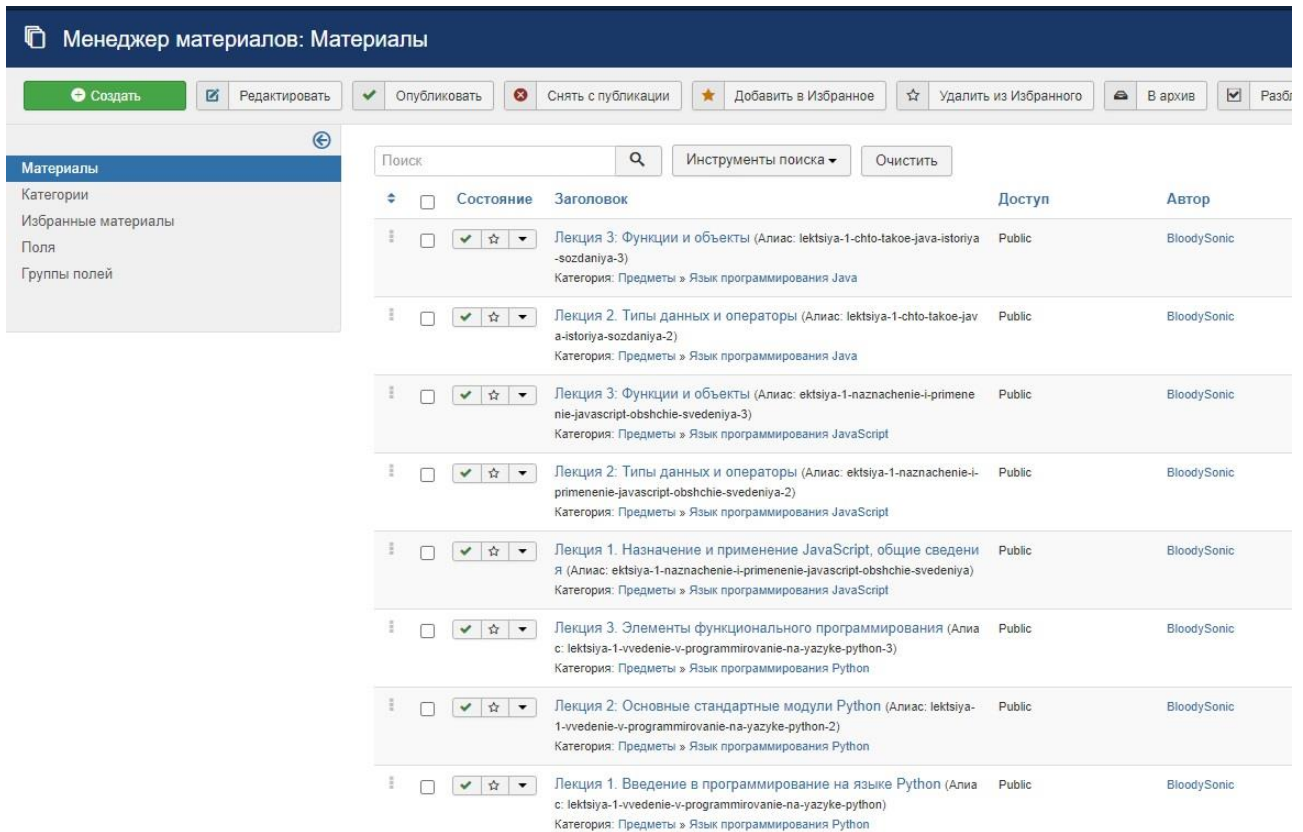


Рисунок 3.2.4 — Матеріали до лекцій

Настройка групп вопросов, все изменения вступают в силу без сохранения теста!

### Создать новую группу вопросов

**Лекция 1 Java тест**

редактировать  удалить

Название группы:

Заголовок группы:

Показывать заголовки:

Описание:

Показывать описание:

Пройходной балл группы:

Количество вопросов:

Выбор вопросов:

Возврат к вопросам:

### Создать вопрос

Что будет выведено на консоль при вычислении следующего выражения?

Изменить    Показать ответы

---

Что будет результатом выполнения программы?

Изменить    Показать ответы

---

Что напечатает следующий фрагмент кода:

Изменить    Показать ответы

---

Что будет выведено на консоль при вычислении следующего выражения?

Изменить    Показать ответы

---

Что произойдет при попытке скомпилировать и запустить данный код? Считается, что файл output.txt уже создан.

Изменить    Показать ответы

---

Какие из следующих строк скомпируются без ошибок?

Изменить    Показать ответы

Рисунок 3.2.5 — Створення тестування

<input type="checkbox"/>	Состояние	Заголовок	Позиция ^
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Обратная Связь!	Нет
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Категории	Нет
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Тесты	Нет
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	О нас	bottom1
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Обучение	bottom2
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Наши друзья!	bottom3
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Быстрый переход	bottom4
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Breadcrumbs	breadcrumb
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Поиск	left
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Latest News	left
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Information	left
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Off Canvas Menu	offcanvas
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Поиск	offcanvas
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Portfolio Module	pagebuilder
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Portfolio Module - Portfolio Home	pagebuilder
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Поиск	right
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	FontSIZE	right
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Последние материалы	right
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Узнайте больше!	right

Рисунок 3.2.6 — Список використуваних модулів

## ВИСНОВОКИ

Розглянуті питання в дипломній роботі є актуальними в області розробки і створення сучасного Web-ресурсу.

В процесі виконання роботи були вирішені наступні приватні задачі:

- розглянуті і застосовані сучасні технології створення Web-додатків;
- вивчено сучасне програмне забезпечення, що застосовується для створення Web-додатків;
- розглянуті різні методи і способи розміщення на web-сторінках різноманітної інформації;
- розглянуті і застосовані основні правила і рекомендації по розробці і створенню Web-сайтів;
- визначена структура і вміст Web-сторінок для сайту;
- створена база даних для Web-додатку;
- проведено аналіз працездатності Web-додатку.

В результаті вивчення даного напрямку була вибрана мова програмування php, розроблені таблиці стилів для дизайну сайту, створено сучасний Web-додаток для навчання студентів «Studmeplease», сайт наповнений контентом.

Розроблений сайт має ряд відмінних рис:

- сучасний дизайн;
- гнучка система настройки відображення інформації;
- кроссбраузерність розробленого сайту;
- розроблено модуль реєстрації та пошуку.
- розроблено модуль зворотного зв'язку.

## СПИСОК ЛІТЕРАТУРИ

1. Барысов Р. И. Постройте профессиональный сайт сами. - СПб., 2013. -304 с.
2. Сухов К. HTML5 – путівник по технології. – М.: ДМК Пресс, 2013. – 352 с.
3. Колисниченко Д.Н. Joomla! Руководство пользователя. - М.: Диалектика, 2013. - 256 с.
4. Джон Рейсинг – “JavaScript Професіональні прийоми програмування” – изд. Питер, 2008. – 352с.
5. Jake Alurlock “Bootstrap” – Publisher: O'Reilly Media, Inc. [електронний ресурс], Режим доступу:  
<https://www.oreilly.com/library/view/bootstrap/9781449344573/>
6. jQuery Step by Logicalm Step by Nickolas Johnson – [електронний ресурс], Режим доступу: <http://nicholasjohnson.com/jquery-book/>
7. Learn PHP hypertext preprocessor – [електронний ресурс],  
Режим доступу: [https://www.tutorialmaloint.com/php/php\\_introduction.htm](https://www.tutorialmaloint.com/php/php_introduction.htm)
8. What is MySQL? – An Introduction To Database Management Systems – [електронний ресурс], Режим доступу:<https://www.edureka.co/blog/what-is-mysql/>
9. Уоринг, Э. Joomla 2.5 / пер. с англ. – СПб.: Питер, 2011. - 180 с.
10. Патерн Model-View-Controler – [електронний ресурс], Режим доступу: <https://www.edureka.co/blog/what-is-mysql/>
11. Локальний сервер WAMP – [електронний ресурс], Режим доступу:  
<https://techterms.com/definition/wamp>



## ДОДАТКИ

### Додаток А

#### Власний заданий шаблону стиль веб-сайта(Preset):

```

ppb-rowa {
  color: #0f9ed6;
}
a:hover {
  color: #0c7ba6;
}
.al-megamenu-parent >li.active>a,
.al-megamenu-parent >li:hover>a {
  color: #0f9ed6;
}
.al-megamenu-parent .al-dropdown li.al-menu-item >a:hover {
  background: #0f9ed6;
}
.al-megamenu-parent .al-dropdown li.al-menu-item.active>a {
  color: #0f9ed6;
}
#offcanvas-toggler >i {
  color: #000000;
}
#offcanvas-toggler >i:hover {
  color: #0f9ed6;
}
.btn-primary,
.alm-btn-primary {
  border-color: #0d89b9;
  background-color: #0f9ed6;
}
.btn-primary:hover,
.alm-btn-primary:hover {
  border-color: #0a698f;
  background-color: #0c7ba6;
}
ul.socialm-icons >li a:hover {
  color: #0f9ed6;
}
.al-page-title {
  background: #0f9ed6;
}
.layout-boxed .body-innerwrapper {
  background: #ffffff;
}

```

```

.al-module ul >li >a {
  color: #000000;
}
.al-module ul >li >a:hover {
  color: #0f9ed6;
}
.al-module .latestnews >div >a {
  color: #000000;
}
.al-module .latestnews >div >a:hover {
  color: #0f9ed6;
}
.al-module .tagscloud .tag-name:hover {
  background: #0f9ed6;
}
.al-simpleportfolio .al-simpleportfolio-filter > ul > li.active >
a,
.al-simpleportfolio .al-simpleportfolio-filter > ul > li > a:hover
{
  background: #0f9ed6;
}
.search .btn-toolbar button {
  background: #0f9ed6;
}
.post-format {
  background: #0f9ed6;
}
.voting-symbol alan.star.active:before {
  color: #0f9ed6;
}
.al-rating alan.star:hover:before,
.al-rating alan.star:hover ~ alan.star:before {
  color: #0c7ba6;
}
.entry-link,
.entry-quote {
  background: #0f9ed6;
}
blockquote {
  border-color: #0f9ed6;
}
.carousel-left:hover,
.carousel-right:hover {
  background: #0f9ed6;
}
#al-footer {
  background: #0f9ed6;
}
.al-comingsoon body {

```

```

        background-color: #0f9ed6;
    }
    .pagination>li>a,
    .pagination>li>alan {
        color: #000000;
    }
    .pagination>li>a:hover,
    .pagination>li>a:focus,
    .pagination>li>alan:hover,
    .pagination>li>alan:focus {
        color: #000000;
    }
    .pagination>.active>a,
    .pagination>.active>alan {
        border-color: #0f9ed6;
        background-color: #0f9ed6;
    }
    .pagination>.active>a:hover,
    .pagination>.active>a:focus,
    .pagination>.active>alan:hover,
    .pagination>.active>alan:focus {
        border-color: #0f9ed6;
        background-color: #0f9ed6;
    }
}

```

### Настройка стилю компонента тестування:

```

.filter-bar, .data-bar
{float:left;clear:left;display:block;position:relative;padding:5px 0 5px
0;width:100%;}
#editcell
{float:left;clear:left;display:block;position:relative;width:100%;}
.filter-search
{float:left;clear:left;display:block;position:relative;width:100%;}
a.disabled {visibility:hidden;}
textarea.tfbtextarea {width:90%;height:50px;}
.tfbhidden {visibility:hidden;height:0px;}
.tfbshow {visibility:visible;height:auto;}
.row0 {
background-color: #F9F9F9;
border-top: 1px #DDDDDD solid;
border-bottom: 1px #DDDDDD dotted;
}

```

```
.row1 {
background-color: #ffffff;
border-top: 1px #DDDDDD solid;
border-bottom: 1px #DDDDDD dotted;
}
.jr_margin_button {
margin: 10px 10px 0 0;
}
.jr_container {
float:left;
clear: left;
display: block;
width:100%;
}
.jr_top {
-moz-border-bottom-colors: none;
-moz-border-left-colors: none;
-moz-border-right-colors: none;
-moz-border-top-colors: none;
border-color: -moz-use-text-color -moz-use-text-color #E5E5E5;
border-image: none;
border-style: none none solid;
border-width: 0 0 1px;
color: #333333;
font-size: 19.5px;
line-height: 36px;
margin-bottom: 18px;
width: 100%;
float:left;
clear: left;
display: block;
}
.jr_bottom {
padding: 18px 0 10px 0;
width: 100%;
float:left;
clear: left;
display: block;
```

```
position: relative;
}
.jr_left {
float:left;
clear: left;
display: block;
width:58%;
padding:1%;
}
.jr_right {
float:left;
display: block;
width:38%;
padding:1%;
}
.jr_left_side {
float:left;
clear: left;
display: block;
width:48%;
padding:1%;
}
.jr_right_side {
float:left;
display: block;
width:48%;
padding:1%;
}
.jr_row {
float:left;
clear:left;
display: block;
position: relative;
}
.jr_label {
float:left;
display: block;
position: relative;
```

```

color:#0088CC;
font-weight:bold;
}
.jr_field {
float:left;
display: block;
position: relative;
}

```

**Фрагмент коду компонента тестування з боку панелі адміністратора створення запитання: Questionadd.php**

```

// Get post
$question_id = $_POST['question_id'];
$verification = $_POST['verification'];
$http_referer = $_POST['http_referer'];
if ($verification == $verification_code) {
// Get question
$db->setQuery( 'SELECT * FROM `#__jr_tfb_question` WHERE id =
' . $question_id );
$question_data = $db->loadObject();
// Get question
echo '
<div id="answer">';
?>
<div class="jr_row jr_width_97 jr_block_new_form">
<input type="hidden" name="edit_question" value="<?php echo
$question_id; ?>" />
<div class="jr_row jr_width_100">
<div class="jr_label_field jr_width_30 hasTooltip" title="<?php
echo JText::_ ( 'JR_TFB_TEST_QUESTION_NAME_DESC' ); ?>">
<?php echo JText::_ ( 'JR_TFB_TEST_QUESTION_NAME' ); ?>
</div>
<div class="jr_field jr_width_70">
<textarea name="edit_question_name" class="jr_width_97"
rows="5"><?php echo $question_data->question_title; ?></textarea>
</div>
</div>
<div class="jr_row jr_width_100">
<div class="jr_label_field jr_width_30 hasTooltip" title="<?php
echo JText::_ ( 'JR_TFB_TEST_QUESTION_TYPE_DESC' ); ?>">
<?php echo JText::_ ( 'JR_TFB_TEST_QUESTION_TYPE' ); ?>
</div>
<div class="jr_field jr_width_70">

```

```

<select name="edit_question_type" class="jr_width_97">
<option value="0"<?php if ($question_data->question_answer == 0)
echo ' selected = "selected"'; ?><?php echo JText::_ (
'JR_TFB_SELECT' ); ?></option>
<option value="1"<?php if ($question_data->question_answer == 1)
echo ' selected = "selected"'; ?><?php echo JText::_ (
'JR_TFB_RADIO' ); ?></option>
<option value="2"<?php if ($question_data->question_answer == 2)
echo ' selected = "selected"'; ?><?php echo JText::_ (
'JR_TFB_CHECKBOX' ); ?></option>
</select>
</div>
</div>

```

```

<div class="jr_row jr_width_100">
<div class="jr_label_field jr_width_30 hasTooltip" title="<?php
echo JText::_ ( 'JR_TFB_TEST_QUESTION_ANSWER_COLUMNS_DESC' ); ?>">
<?php echo JText::_ ( 'JR_TFB_TEST_QUESTION_ANSWER_COLUMNS' ); ?>
</div>
<div class="jr_field jr_width_70">
<input type="text" name="edit_question_answer_columns"
class="jr_width_10" value="<?php echo $question_data-
>question_answer_columns; ?>" />
</div>
</div>

```

### **Фрагмент коду компонента тестування з боку панелі адміністратора створення**

#### **відповідей: Answersadd.php**

```

echo '
<div id="new_answer">
<div class="jr_row jr_width_97 jr_block_new_form"
id="jr_block_new_group">
<div class="jr_row jr_width_100">
<div class="jr_label_field jr_width_30 hasTooltip" title="'.JText::_ (
'JR_TFB_TEST_ANSWER_NAME_DESC' ).'">
'.JText::_ ( 'JR_TFB_TEST_ANSWER_NAME' ) .'
</div>
<div class="jr_field jr_width_70">
<textarea name="answer_name" class="jr_width_95" rows="5"></textarea>
</div>
</div>
<div class="jr_row jr_width_100">

```

```

<div class="jr_label_field jr_width_30 hasTooltip" title="'.JText::_(
'JR_TFB_TEST_ANSWER_RIGHT_DESC' ).'">
'.JText::_( 'JR_TFB_TEST_ANSWER_RIGHT' ).'
</div>
<div class="jr_field jr_width_70">
<select name="answer_right" class="jr_width_97">
<option value="0">'.JText::_( 'JNO' ).'</option>
<option value="1">'.JText::_( 'JYES' ).'</option>
</select>
</div>
</div>
<div class="jr_row jr_width_100">
<div class="jr_label_field jr_width_30 hasTooltip" title="'.JText::_(
'JR_TFB_TEST_ANSWER_POINT_DESC' ).'">
'.JText::_( 'JR_TFB_TEST_ANSWER_POINT' ).'
</div>
<div class="jr_field jr_width_70">
<input type="text" name="answer_point" class="jr_width_50" value="0" />
</div>
</div>
<div class="jr_row jr_width_100">
<div class="btn btn-small btn-success answer_add" rel="'. $question_id.'">
'.JText::_( 'JR_TFB_ADD' ).'
</div>
<div class="btn btn-small btn-remove" id="answer_cancel">
'.JText::_( 'JR_TFB_CANCEL' ).'
</div>
</div>
</div>
</div>
';
}
else echo '
<div id="new_answer">failure</div>';

```



**Фрагмент коду компонента тестування з боку користувача: Question.php**

```

/ Get questions 1-1|2-1,2,3|3-1,2,3          id групи - id вопроса, id
вопроса |
// Create first group id
$test_group_first = 0;
foreach ($test_groups as $test_groups_item) {
// Create first group id
if ($test_group_first == 0) $test_group_first = $test_groups_item->id;
$db->setQuery( 'SELECT group_back FROM `#__jr_tfb_group` WHERE id =
' . $test_groups_item->id);
$group_back = $db->loadResult();
// Create groups questions
$test_group[] = $test_groups_item->id;
// Create group questions
${'test_group_questions_' . $test_groups_item->id} = array();

// Get limit of questions
if ($test_groups_item->group_questions == 0 ) $limit = '';
else $limit = ' LIMIT ' . $test_groups_item->group_questions;
// Get questions order
if ($test_groups_item->group_select == 0 ) $ordering = 'ordering';
else $ordering = 'RAND()';
$db->setQuery( 'SELECT id FROM `#__jr_tfb_question` WHERE question_group
= ' . $test_groups_item->id.' ORDER BY ' . $ordering . $limit);
$test_questions_list = $db->loadObjectList();
// Create questions string
if ($test_questions_list) {
if ($test_questions != '') $test_questions .= '|';
$test_questions .= $test_groups_item->id.'-';
$test_question_count = 0;
$question_number = 0;
foreach ($test_questions_list as $test_question) {
// Add first group questions
if ($question_number == 0) $group_first_questions[] = $test_question->id;
// Add first question ID
if ($group_back == 0 OR ($test_data->test_back == 0 AND $question_number
== 0)) $group_back_questions[] = $test_question->id;
// Add question ID to group list

```

```

${'test_group_questions_' . $test_group_data[0]}[] = $test_group_item;
// Add question to test list
$question_number++;
${'test_group_questions_' . $test_groups_item->id}[] = $test_question->id;
if ($test_question_count > 0) $test_questions .= ',';
$test_questions .= $test_question->id;
$test_question_count++;
$test_questions_table[] = $test_question->id;
}
$group_last_questions[] = $test_question->id;
}
}

```

### Фрагмент коду компонента тестування з боку користувача: Test.php

```

// Get test result
$db->setQuery( "SELECT * FROM `#__jr_tfb_result` WHERE id =
".$result_id );
$test_user_item = $db->loadObject();
if ($test_user_item->result_result == 2) {
$j_r_tfb_test_date = JHTML::_('date', $test_user_item->result_date,
'd.m.Y').JText::_('JR_TFB_TEST_TIME_ON').JHTML::_('date',
$test_user_item->result_date, 'h:i');
}
else {
$j_r_tfb_test_date = JText::_('JR_TFB_TEST_STATLIST_NOT_COMPLETED'
);
}
if ($test_user_item->result_passed == 1 AND $test_user_item-
>result_result == 2) {
$filter_line = "filter_line_passed";
}
else if ($test_user_item->result_passed == 0 AND $test_user_item-
>result_result == 2) {
$filter_line = "filter_line_failed";
}
else {
$filter_line = "filter_line";
}
$j_r_tfb_test_time = $test_time_passed = date("H:i:s", mktime(0, 0,
$test_user_item->result_timer));
// Get test
$db->setQuery( "SELECT * FROM `#__jr_tfb_test` WHERE id =
".$test_user_item->result_test );
$test_item = $db->loadObject();

```

```

// Get user forms
$db->setQuery( "SELECT * FROM `#__jr_tfb_form`" );
$test_forms = $db->loadObjectList();
// Get user data
$db->setQuery( "SELECT * FROM `#__jr_tfb_user` WHERE user_user =
".$test_user_item->result_user );
$test_user = $db->loadObject();
$test_user_data = '';
foreach ($test_forms as $test_form) {
$user_field_id = 'user_'.$test_form->id;
if (isset($test_user->$user_field_id) AND $test_user-
>$user_field_id != '') {
if ($test_form->form_type == 3) {
$test_user_field = '';
$test_user_options = explode ("|", $test_user->$user_field_id);
$test_form_options = explode ("%|%", $test_form->form_options);
$test_form_option = 0;
foreach ($test_user_options as $test_user_option) {
if ($test_user_option == 1) {
$test_user_option = $test_form_options[$test_form_option];
$test_user_field .= '<div
class="filter_test_data">'.$test_user_option.'</div>';
}
$test_form_option++;
}
}
else $test_user_field = $test_user->$user_field_id;
$test_user_data .= '
<div class="filter_test_line filter_user_data filter_test_show">
<div class="filter_test_inline jr_tfb_width50
filter_test_label">'.$test_form->form_title.'</div>
<div class="filter_test_inline jr_tfb_width50
filter_test_data">'.$test_user_field.'</div>
</div>';
}
}
$test_user_resume = '';
if ($test_user_item->result_answers == "") $test_user_answers =
'<span class="jr_form_error filter_test_show">'.JText::_ (
'JR_TFB_TEST_STATLIST_NO_ANSWERS' ).'</span>';
// Get groups
$db->setQuery( "SELECT * FROM `#__jr_tfb_group` WHERE group_test =
".$test_user_item->result_test );
$test_groups = $db->loadObjectList();
$test_group = array();
$test_group_list = '';
foreach ($test_groups as $test_groups_item) {
$test_group[$test_groups_item->id] = $test_groups_item-
>group_title;
}

```

```

if ($test_group_list != '') $test_group_list .= ',';
$test_group_list .= $test_groups_item->id;
}
// Get questions and answers
$db->setQuery( "SELECT * FROM `#__jr_tfb_question` WHERE
question_group IN ($test_group_list)" );
$test_questions = $db->loadObjectList();
foreach ($test_questions as $test_questions_item) {
$test_question[$test_questions_item->id] = $test_questions_item-
>question_title;
$test_advice[$test_questions_item->id] = $test_questions_item-
>question_advice;
$test_correct[$test_questions_item->id] = $test_questions_item-
>question_correct;
// Get answers
$db->setQuery( "SELECT * FROM `#__jr_tfb_answer` WHERE
answer_question = ".$test_questions_item->id." ORDER BY ordering"
);
$test_answers = $db->loadObjectList();
${'test_answer_text_'.$test_questions_item->id} = array();
${'test_answer_result_'.$test_questions_item->id} = array();
${'test_answer_correct_'.$test_questions_item->id} = 0;
foreach ($test_answers as $test_answer) {
${'test_answer_text_'.$test_questions_item->id}[$test_answer-
>ordering] = $test_answer->answer_title;
${'test_answer_result_'.$test_questions_item->id}[$test_answer-
>ordering] = $test_answer->answer_correct;
if ($test_answer->answer_correct == 1)
${'test_answer_correct_'.$test_questions_item->id}++;
}
}

```