

Разработка электронного образовательного ресурса «Использование GitHub как системы для непрерывной интеграции и развёртывания современного веб-проекта»

Косыгин Кирилл Сергеевич
Институт информационных технологий и
технологического образования, 4 курс

Актуальность исследования

1. Значимость владения методологией CI/CD для ИТ-специалистов.

[Вакансии](#) [Резюме](#) [Компании](#)

744 вакансии «ci/cd»

[Вакансии](#) [Резюме](#) [Компании](#)

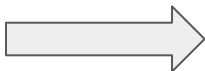
Работа DevOps в Санкт-Петербурге, 524 вакансии

Требования:

Ищем middle, но готовы рассмотреть и junior на других условиях.

- Уверенное знание PHP, JS, CSS, HTML;
- Навыки работы с базами данных MySQL, MariaDB, знание языка SQL;
- Опыт работы с фреймворками (Symfony);
- Умение работать с Git (на уровне использования веток, решения конфликтов пушить, мержить);
- ООП и паттерны программирования;
- Умение работать с различными API;
- Документирование своего кода (PHPDoc);
- Использование CI/CD, Docker, Docker-compose;
- Желание развиваться самостоятельно и в команде

2. Разрозненность существующей информации.



Перспектива внедрения ЭОР в педагогический процесс

Предмет исследования

Применение компьютерных и цифровых технологий в обучении



Разработка электронного образовательного ресурса практической направленности, предназначенного для изучения практик CI/CD в среде GitHub

Цель исследования

Разработать электронный образовательный ресурса «Использование GitHub как системы для непрерывной интеграции и развёртывания современного веб-проекта»

Задачи

1. Изучить сущность понятия ЭОР и особенности их разработки.
2. Провести анализ инструментов, предназначенных для разработки ЭОР.
3. Адаптировать требования к разрабатываемому ЭОР с учётом особенностей изучения методологии CI/CD.
4. Спроектировать ЭОР «Использование GitHub как системы для непрерывной интеграции и развёртывания современного веб-проекта».
5. Создать ЭОР в соответствии с разработанным планом.

Значимость исследования

Теоретическая

Рассматриваются и обобщаются теоретические аспекты разработки ЭОР: основные требования к ЭОР, способы адаптации под специфические требования и типовые инструменты разработки.

Практическая

Разработанный ресурс может быть использован учащимися для освоения практик CI/CD или применён при разработке других учебных материалов.

Проблематика разработки ЭОР

1. Определение и сущность.
2. Особенности применения.
3. Технологические особенности.
4. Требования.
5. Анализ инструментов.


Анализ инструментов

1. LMS-системы: Moodle, ATutor, Sakai, и т. п.
2. CMS-базированные решения: WordPress + [LifterLMS, Sensei, ...], Drupal, и т. п.
3. собственный программный продукт.
4. Прочие инструменты: iSpring Suite, Everest, NeoBook, и т. п.

Разработка ЭОР

Контекст: разработка современного веб-проекта с применением CI/CD

Symfony Demo Application

The Symfony Demo application is designed to **run out-of-the-box**, so you won't need to configure anything or run any other command. Alternatively, you can get the demo application through the [symfony/symfony-demo repository](#) .

What is this application useful for?

1) Learning Symfony

This is the most obvious reason to use this demo application and its main purpose when we developed it.

2) Teaching Symfony

This application can also be a useful teaching resource to train your company interns and to deliver workshops during [events organized by the Symfony community](#).

3) Testing new Symfony features

About

Symfony Demo Application

 [symfony.com/](#)

[php](#) [demo](#) [symfony](#)
[symfony-application](#)

 Readme

 MIT license

 Code of conduct

☆ 2.2k stars

👁 126 watching

🍴 1.6k forks

Releases 82

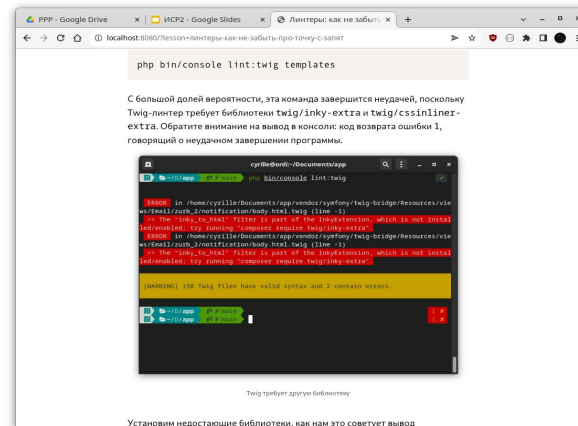
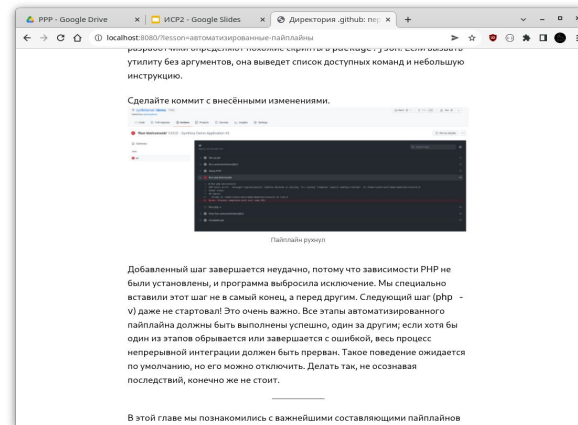
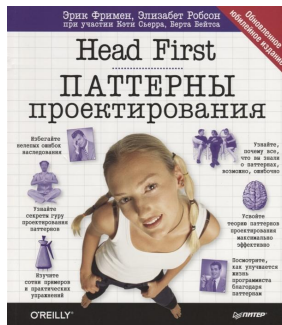
 **2.0.2** Latest
on 21 Feb

[+ 81 releases](#)

Особенности ЭОР

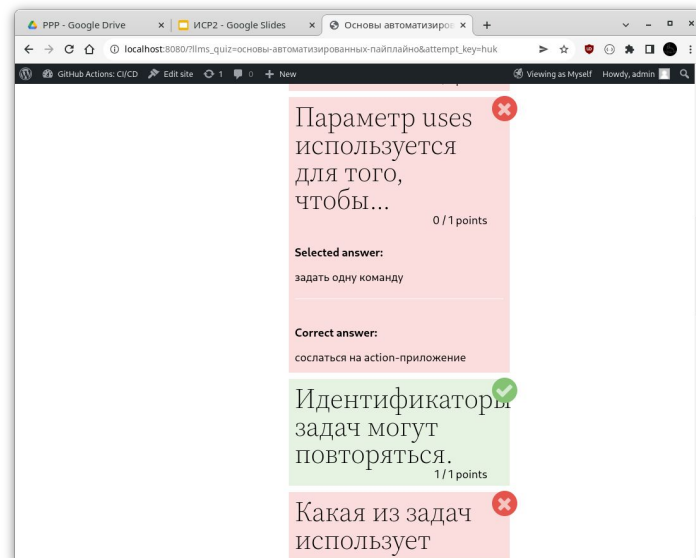
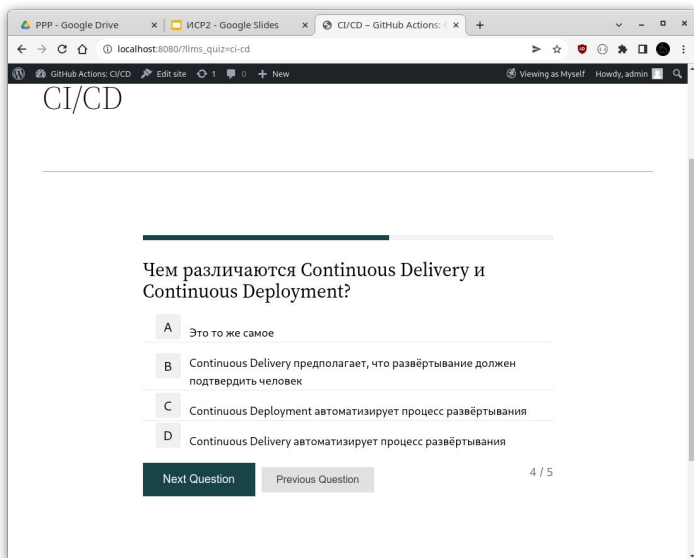
Перенятие принципов “Head First”:

- наглядность;
- разговорный стиль изложения;
- активное участие читателя;
- привлечение и сохранение внимания читателя;
- обращение к эмоциям.



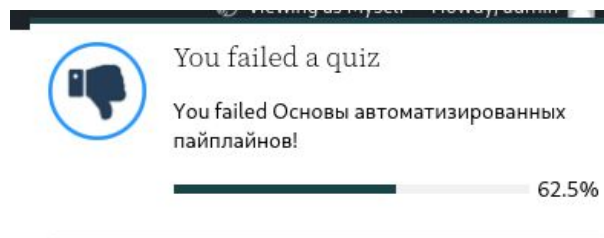
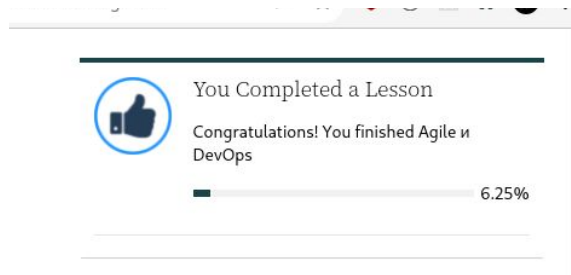
Особенности ЭОР

Самоконтроль



Особенности ЭОР

Геймификация



Структура ЭОР

1. Введение

1. Agile и DevOps.
2. Непрерывная интеграция.
3. Непрерывная доставка и развёртывание.

2. Continuous Integration

1. SSH: Защищённое соединение.
2. Размещение проекта в GitHub.
3. Директория .github: первый пайплайн.
4. Кеш GitHub: ускорение пайплайнов.
5. PHPUnit: автоматизированное тестирование.
6. Линтеры.
7. PHP CS Fixer: стандарты написания кода.
8. PHPStan: типобезопасность без компиляции кода.
9. Symfony CLI: поиск уязвимостей в зависимостях.

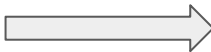
3. Continuous Delivery

1. Подготовка сервера.
2. Rsync: ручная доставка.
3. Deployer: автоматическая доставка и развёртывание.
4. Теги и релизы: порядок в пайплайне.

Выбор технологий

- Удобство редактирования текста;
- возможность создавать викторины;
- скорость разработки;
- простота развёртывания.

YAGNI

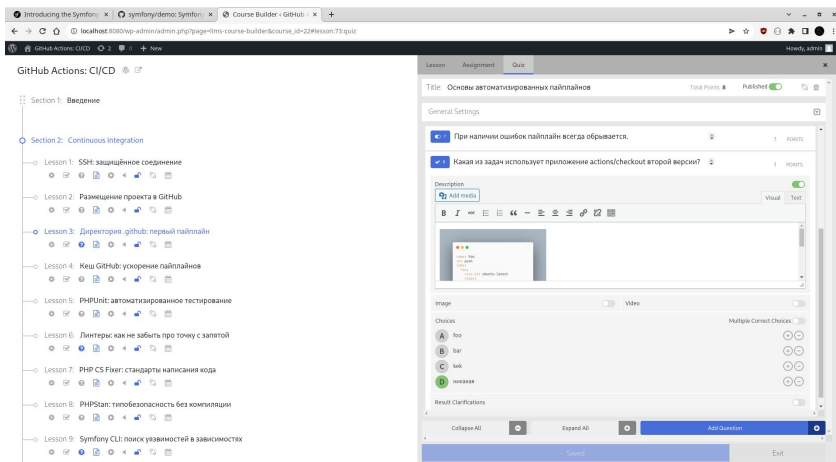


WordPress + LifterLMS, Prismatic

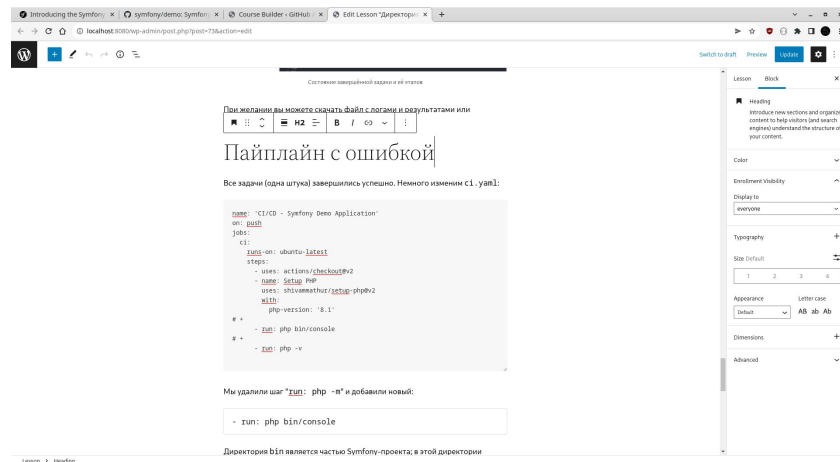
Docker Compose:

- WordPress;
- Apache;
- MySQL.

Создание ЭОР



Редактор курсов и викторин



Текстовый редактор

Развёртывание ЭОР

```
cyrille@ordi:~/Bureau/gha
cd Bureau
git clone git@github.com:cyrillelamar/gha.git
Clonage dans 'gha'...
remote: Enumerating objects: 5550, done.
remote: Total 5550 (delta 0), reused 0 (delta 0), pack-reused 5550
Réception d'objets: 100% (5550/5550), 41.41 Mio | 2.80 Mio/s, fait.
Résolution des deltas: 100% (1416/1416), fait.
cd gha
dcc up -d
Creating network "gha_default" with the default driver
Creating gha_db_1 ... done
Creating gha_wordpress_1 ... done
```

Git Flow by Vincent Driessen

Практика сводится к наличию нескольких типов веток и тому, как они сливаются:

- master (main) – собственно релизы, которые видят заказчики;
- release – стабильные версии приложения, потенциальные релизы;
- develop – нестабильная версия приложения, находящаяся в активной разработке;
- feature – ветки с новым функционалом;
- hotfix – ветки с экстренными исправлениями.

Обратите внимание на то, что релиз принадлежит мастеру: voto в нескольких

Выводы

Поставленные задачи были реализованы:

1. Изучена сущность, классификация и особенности разработки ЭОР.
2. Проведён анализ инструментов, предназначенных для разработки ЭОР.
3. Требования к ЭОР адаптированы с учётом особенностей изучения практик CI/CD.
4. Спроектирован ЭОР «Использование GitHub как системы для непрерывной интеграции и развёртывания современного веб-проекта».
5. В соответствии с планом создан ЭОР «Использование GitHub как системы для непрерывной интеграции и развёртывания современного веб-проекта» с применением технологий WordPress, MySQL, Docker (Compose), Git.