**6** РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

**6.1** Установка системы на удаленном сервере

Для написания веб-приложения использовался фреймворк Ruby on Rails, а также язык Ruby. Это накладывает некоторые ограничения на использование операционной системы. Проблема в том, что во многих плагинах используются нативные инструменты семейства Linux, что станет проблемой при использовании на Windows. Поэтому для развертывания был выбран сервер с операционной системой Ubuntu 14.04 LTS.

Данное веб-приложение не предъявляет высоких требований к возможностям сервера, но для стабильной работы требуется следующий сервер:

* 2-х ядерный процессор Intel/AMD;
* 2 Гб оперативной памяти;
* желательно наличие SSD;
* cетевое подключение 10 Мб/с

После установки операционной системы, необходимо установить все необходимые пакеты для функционирования системы и зависимости к ним:

* RVM
* Git
* MySQL
* Node.js
* Elasticsearch
* Java JDK
* Nginx

Для начала необходимо установить и настроить среду, в которой будет исполняться язык Ruby.

Для выбранной операционной системы можно поставить пакет для языка Ruby при помощи пакетного менеджера, но версия Ruby не будет актуальной. Более предпочтительным вариантом будет установка RVM (Ruby Version Manager). Для установки необходимо выполнить следующую команду:

curl -sSL https://get.rvm.io | bash -s stable

Далее при помощи RVM необходимо установить Ruby-интерпретатор:

rmv install ruby-2.2.5

Для установки Git выполним следующую команду:

sudo apt–get install git

Далее установим Node.js:

sudo apt–get install nodejs

Установим MySQL. Он есть в репозиториях Ubuntu и разбит на несколько пакетов.

Для установки MySQL Сервер необходимо выполнить команду:

sudo apt-get install mysql-server

При установке конфигурационный скрипт запросит пароль администратора(root) для базы данных.

Для установки MySQL Client необходимо выполнить следующую команду:

sudo apt-get install mysql-client

Для установки Php-модуля для MySQL необходимо выполнить:

sudo apt-get install php5-mysql

Далее необходимо выполнить настройку MySQL. Конфигурация сервера MySQL содержится в файле /etc/mysql/my.cnf.

Настроим доступ к серверу из сети. По умолчанию сервер MySQL принимает соединения только с локальной машины. Для того, чтобы разрешить подключаться к нему с других машин замените строку:

bind-address = 127.0.0.1

#разрешить подключатся с любого хоста

bind-address = 0.0.0.0

#разрешить подключатся только с указанного IP

bind-address = 192.168.1.23

Далее необходимо выполнить настройку кодировки. По умолчанию в Ubuntu MySQL устанавливается с кодировкой latin1. В этом можно убедиться посмотрев вывод запроса:

SHOW VARIABLES LIKE 'char%';

Результат выполнения команды:

character\_set\_client latin1

character\_set\_connection latin1

character\_set\_database latin1

character\_set\_filesystem binary

character\_set\_results latin1

character\_set\_server latin1

character\_set\_system utf8

character\_sets\_dir /usr/share/mysql/charsets/

Для изменения кодировки выполним команду:

SET names utf8;

Чтобы кодировка utf8 использовалась по умолчанию для создания таблиц выполним:

DEFAULT CHARACTER SET utf8 COLLATE utf8\_general\_ci;

Чтобы сервер сразу загружался с нужной кодировкой, необходимо отредактировать файл /etc/mysql/my.cnf:

sudo nano -w /etc/mysql/my.cnf

В секцию [mysqld] добавьте следующие строки:

skip-character-set-client-handshake

character-set-server = utf8

init-connect='SET NAMES utf8'

collation-server=utf8\_general\_ci

Так же желательно установить кодировку для клиента и mysqldump. Для этого в секциях [client] и [mysqldump] необходимо добавить строчку:

default-character-set=utf8

Чтобы изменения вступили в силу, перезагружаем сервер MySQL:

sudo service mysql restart

После этого список переменных будет выглядеть так:

character\_set\_client utf8

character\_set\_connection utf8

character\_set\_database utf8

character\_set\_filesystem binary

character\_set\_server utf8

character\_set\_test utf8

character\_set\_development utf8

character\_set\_server utf8

character\_set\_client utf8

character\_set\_system utf8

character\_sets\_dir /usr/share/mysql/charsets/

Установим Elasticsearch, который необходим для полнотекстового поиска. Для начала установим Java JDK:

sudo apt-get -y install openjdk-7-jdk

mkdir /usr/java

ln -s /usr/lib/jvm/java-7-openjdk-amd64 /usr/java/default

Далее продолжаем установку Elasticsearch, скачиваем, распаковываем и запускаем установщик из архива:

cd /opt

wget <https://download.elastic.co/elasticsearch/elasticsearch/elasticsearch-1.7.2.deb>

sudo dpkg -i elasticsearch-1.7.2.deb

java -jar start.jar

После этого пакет Elasticsearch будет установлен в каталог /usr/share/elasticsearch, конфигурационные файлы будут помещены в /etc/elasticsearch, а скрипты – в etc/init.d/elasticsearch.

Далее необходимо выполнить его настройку, для этого нужно открыть файл /etc/elasticsearch/elasticsearch.yml и вставить туда следующие настройки:

node.master: false

index.number\_of\_shards: 1

index.number\_of\_replicas: 0

path.data: /media/different\_media

Таже необходимо поставить запуск Elasticsearch на автостарт:

sudo update-rc.d elasticsearch defaults

После настроики Elasticsearch, необходимо его запустить:

sudo service elasticsearch start

Для функционирования веб-приложения необходим веб-сервер, который обрабатывает запросы извне и отсылает веб-приложению. Для данной задачи используется Nginx.

Для установки Nginx, в команде требуется ввести команду:

sudo apt-get install nginx

Далее необходимо отредактировать файл конфигурации Nginx /etc/nginx/nginx.conf следующими значениями:

upstream app {

# Path to Puma SOCK file, as defined previously

server unix: app\_dir/shared/tmp/sockets/puma.sock fail\_timeout=0;

}

server {

listen 80;

server\_name record.pro;

root app\_dir/current/public;

try\_files $uri/index.html $uri @app;

location @app {

proxy\_pass http://app;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto http;

proxy\_set\_header Host $http\_host;

proxy\_redirect off;

proxy\_next\_upstream error timeout invalid\_header http\_502;

}

location ~\* ^/assets/ {

expires max;

add\_header Cache-Control public;

add\_header Last-Modified "";

add\_header ETag "";

break;

}

error\_page 500 502 503 504 /500.html;

client\_max\_body\_size 4G;

keepalive\_timeout 10;

}

Также необходимо произвести настройку самого веб-приложения. Настроим соединение к базе данных, изменив config/database.yml файл следующим образом:

default: &default

adapter: mysql2

encoding: utf8

pool: 5

username: root

password:

socket: /var/run/mysqld/mysqld.sock

test:

secret\_key\_base: <%= ENV["secret\_key\_base"]>

production:

<<: \*default

database: clear\_production

username: clear

password: <%= ENV['DATABASE\_PASSWORD'] %>

Также необходимо в папке config создать файл secrets.yml и разместить там следующее:

production:

secret\_key\_base: <%= ENV["SECRET\_KEY\_BASE"] %>

Также необходимо в папке envitonments создать файл production.rb и разместить там следующее:

Rails.application.configure do

config.cache\_classes = true

config.serve\_static\_files = true

config.i18n.fallbacks = true

config.assets.digest = true

config.active\_support.deprecation = :notify

config.log\_formatter = ::Logger::Formatter.new

end

Теперь нужно установить плагины, необходимые для работы приложения. Для это в корне веб-приложение выполняем:

bundle install

Далее нужно создать базу данных и провести миграции:

bundle exec rake db:setup

Теперь можно запустить сервер:

sudo service nginx restart

**6.2** Описание работы с web-сервисом

В данном разделе представлено описание всех возможностей программного средства для всех ролей пользователей в системе. Кроме того, описано подробное руководство с пошаговой инструкцией по использования web-сервиса.

В системе есть следующие роли:

* администратор;
* администратор организации;
* пользователь.

**6.2.1** Домашняя страница web-сервиса

При открытии web-сервиса в браузере, для пользователя отображается страница приветствия. Вверху страницы расположены логотип сервиса и навигационное меню. Пользователь может перейти к спискам организаций и курсов или на страницу авторизации.

Кроме того, на этой странице пользователь видит строку для ввода поиска курсов (см. рисунок 6.1). Даже если пользователь не аутентифицирован, он может начать пользоваться сервисом – ввести в строку поиска название интересующего курса, и, если в системе имеются публичный курсы, удовлетворяющие условиям поиска, то сервис вернет их для пользователя.

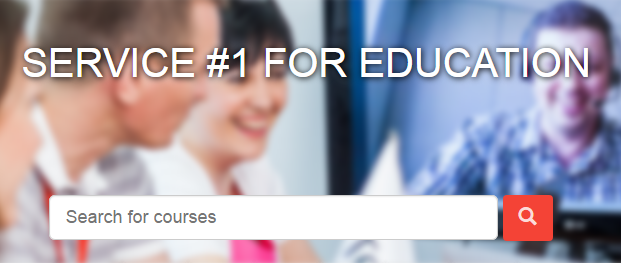


Рисунок 6.1 – Строка для ввода поиска курсов

Внизу страницы расположена краткая информация, описывающая данный сервис и карта зарегистрированных пользователей – рисунок 6.2. С помощью ее можно увидеть пользователи каких стран пользуются данным сервисом наиболее активно.



Рисунок 6.2 – Карта пользователей сервиса

**6.2.2** Вход в личный кабинет

Если пользователь уже имеет зарегистрированный аккаунт в системе, он может войти в свой личный кабинет используя логин и пароль – рисунок 6.3.

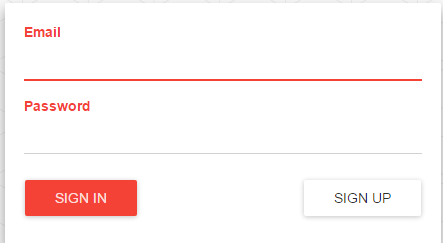


Рисунок 6.3 – Аутентификация пользователя

При успешной авторизации отобразится личный кабинет пользователя, представленный на рисунке 6.4, иначе в верхнем левом углу появится всплывающее сообщение с ошибкой, сообщение с ошибкой представлено на рисунке 6.5.

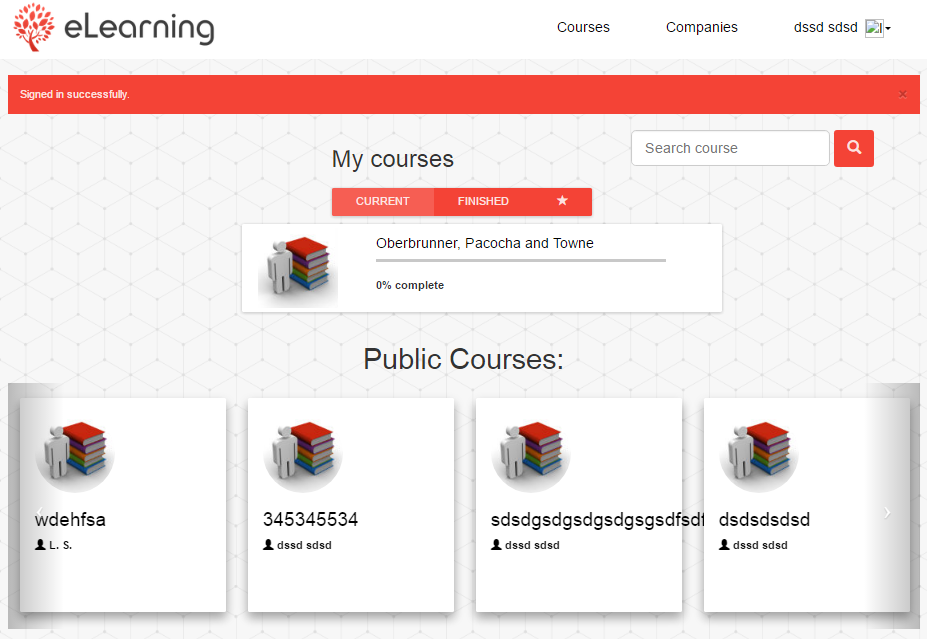


Рисунок 6.4 – Личный кабинет пользователя

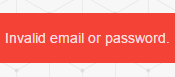


Рисунок 6.5 – Сообщение о неправильном вводе логина или пароля

Если пользователь забыл свой пароль, он может перейти на страницу восстановления пароля (см. рисунок 6.6) и ввести свой емэйл, после чего ему на почту придет сообщением с инструкцией по восстановлению пароля.

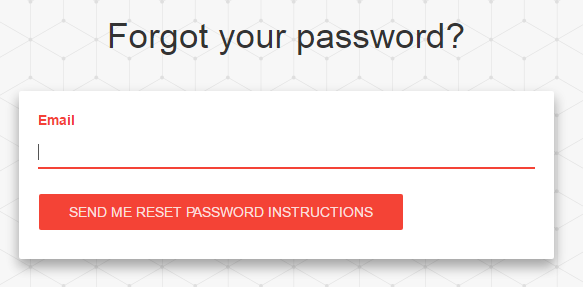


Рисунок 6.6 – Страница восстановления пароля

**6.2.3** Регистрация нового пользователя

Если пользователь не имеет аккаунта в сервисе, он может создать его на странице регистрации, кликнув на кнопку «Sign Up». Пользователю будет представлена форма регистрации нового пользователя (см. рисунок 6.7).

На данной странице необходимо заполнить обязательные для заполнения поля:

* First name;
* Last name;
* Email;
* Password;
* Password confirmation.

Следующие поля на данной странице являются необязательными для заполнения:

* Country;
* Your can choose organization(s).

После заполнения обязательных полей следует нажать кнопку «SIGN UP».

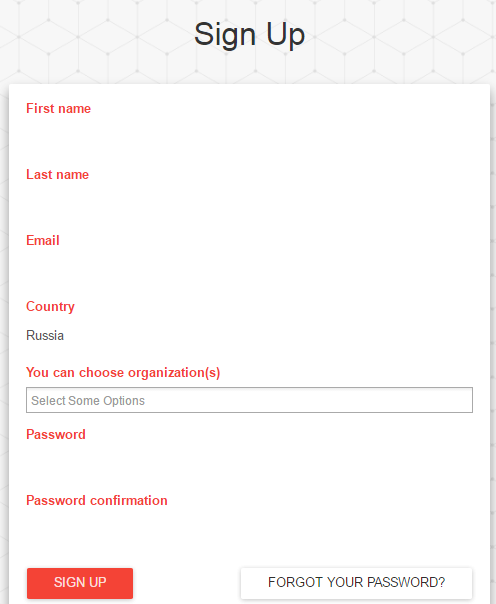


Рисунок 6.7 – Форма регистрации нового пользователя

После завершения процедуры регистрации, web-сервис получает необходимую пользовательскую информацию, на основании её он формирует все необходимые пользовательские модели данных и перенаправляет пользователя на страницу личного кабинета.

**6.2.4** Редактирование личного профиля

После успешной регистрации, пользователь может добавить остальную информацию о себе не странице редактирования личного профиля – рис. 6.8.

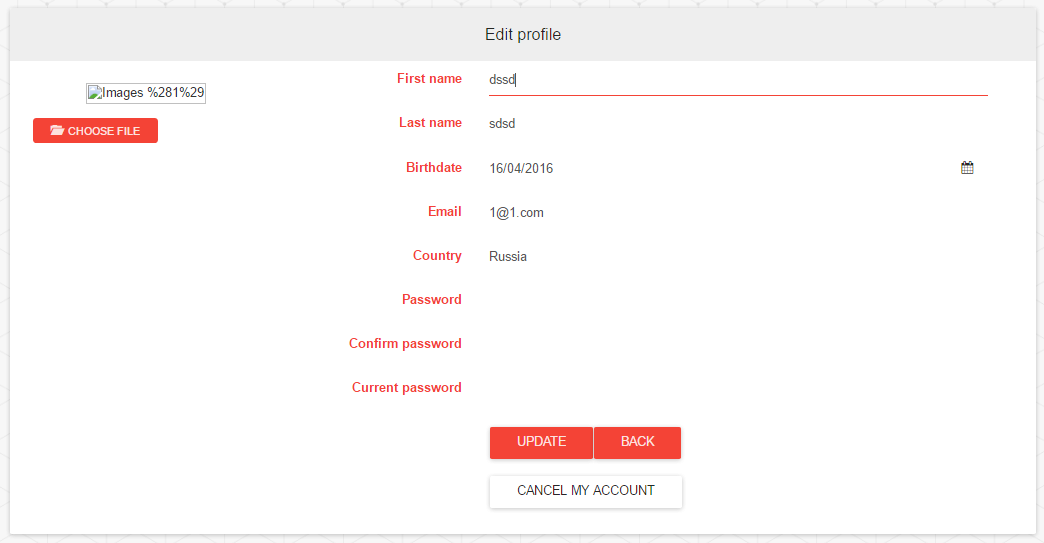


Рисунок 6.8 – Страница редактирования личного профиля

На странице редактирования личного профиля пользователю доступны следующие поля для изменения:

* First name – имя пользователя;
* Last name – фамилия пользователя;
* Birthday – дата рождения пользователя;
* Email – емэйл пользователя;
* Country – страна проживания пользователя;
* Password - новый пароль пользователя;
* Avatar – аватар пользователя.

Кроме того, на этой странице пользователь может удалить свой аккаунт, нажав на кнопку «CANCEL MY ACCOUNT»

**6.2.5** Создание курса

Для создания нового курса необходимо перейти на страницу личного кабинета пользователя и нажать на вкладку «New Course», в результате отобразится форма для заполнения основных данных о новом курсе, представленная на рисунке 6.9.

Для создания курса необходимо заполнить следующие обязательные поля:

* Title – название курса;
* Permission – режим доступа к курсу;
* Image – логотип курса;
* Certificate – сертификат курса (опционально).

Для, необходимо нажать кнопку «Create».

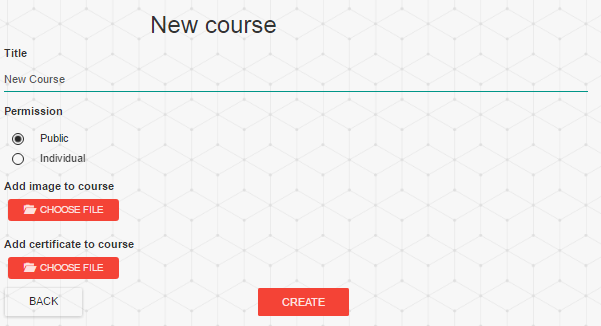


Рисунок 6.9 – Форма создания нового курса

При успешном сохранении курса, пользователь будет перенаправлен на страницу заполнения материалов курса (см. рисунок 6.10), в противном случае пользователю будет показано сообщение об ошибке, представленное на рисунке 6.11.

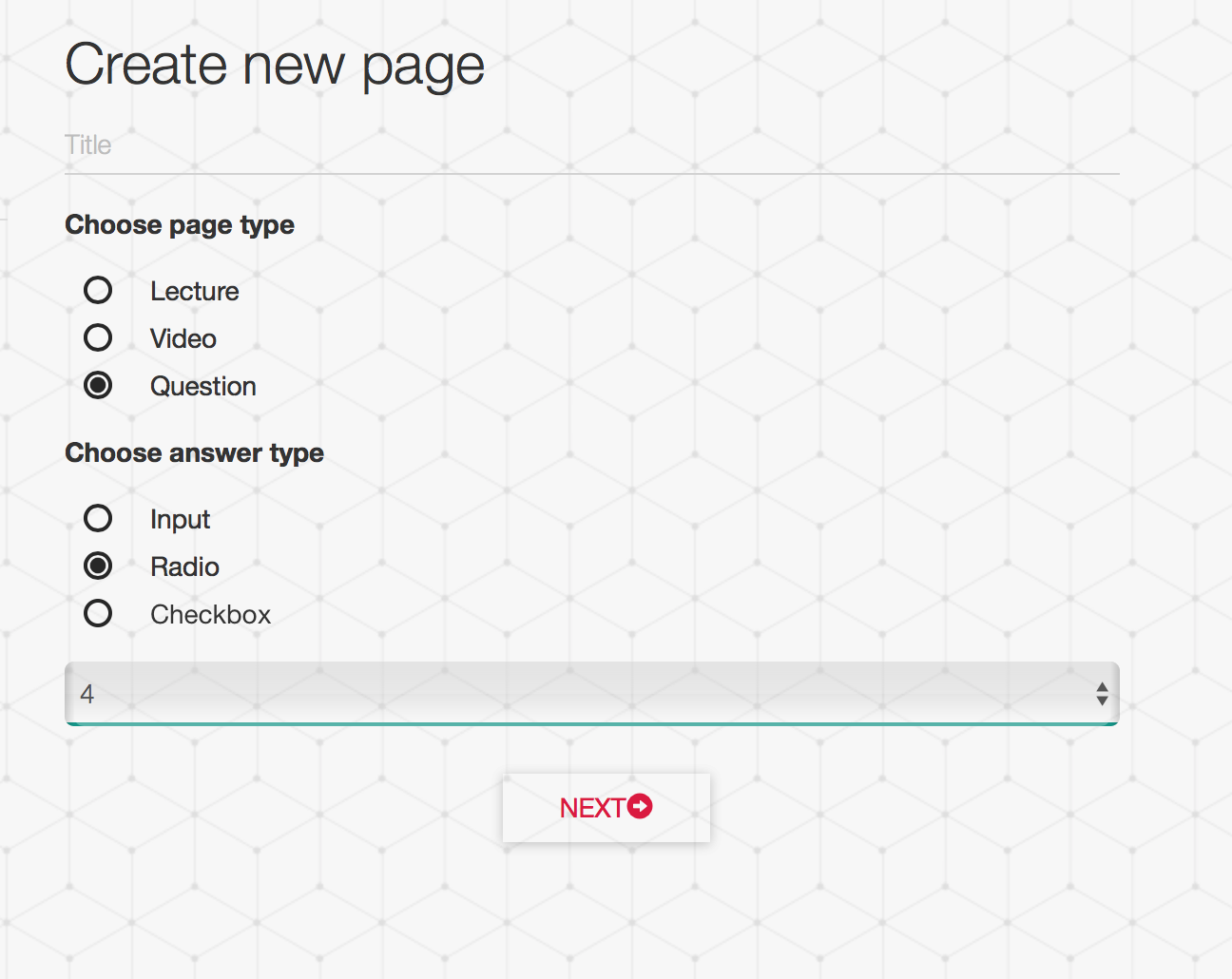


Рисунок 6.10 – Страница заполнения материалов курса

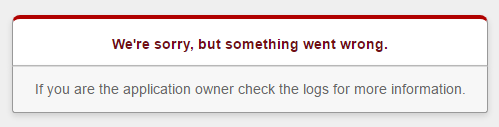


Рисунок 6.11 – Сообщение об ошибке при неудачном сохранении курса

Каждый курс представляет собой набор материалов (лекции, видеозаписи, теоретические опросы), которые в дальнейшем будут использоваться для обучения и контроля знаний. Так, к примеру, процесс создания лекции оснащен редактором WYSIWYG, позволяющий создавать материалы лекций в наиболее удобной для пользователя форме (рис.6.12).

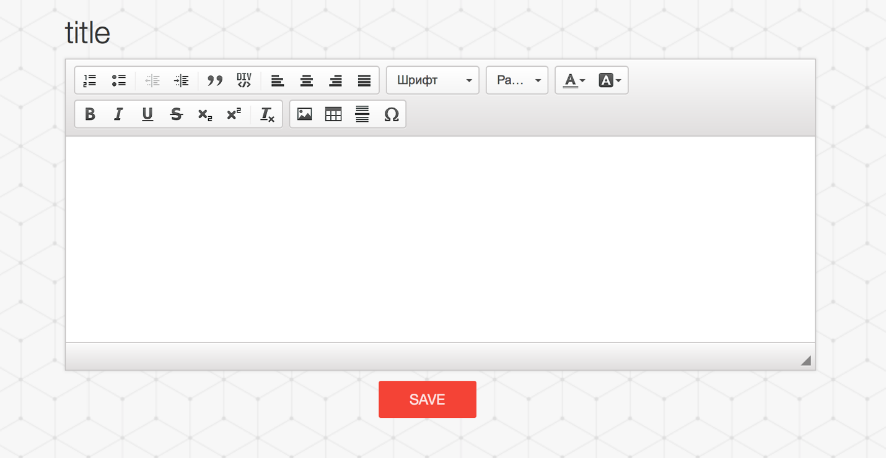


Рисунок 6.12 Страница добавления лекции

**6.2.5** Прохождение курса

Для начала прохождения курса необходимо нажать кнопку «Start Course», будут постранично отображаться все материалы курса – рисунок 6.12.

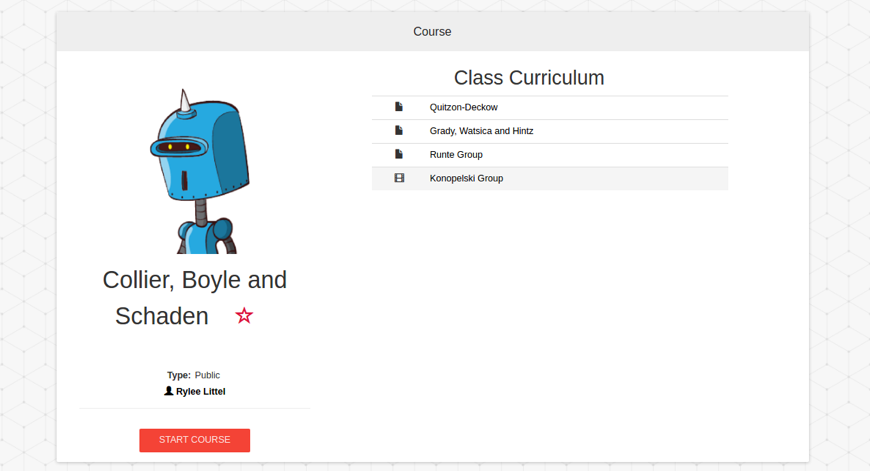


Рисунок 6.13 – Страница прохождения курса

По окончании курса пользователь может получить именной сертификат, шаблон которого прикреплялся при создании курса, его можно скачать в формате PDF (рис. 6.14). В нем содержится следующая информация:

* пользователь – имя и фамилия пользователя прошедшего курс;
* курс – название курса;
* дата – дата окончания прохождения курса;
* автор – имя и фамилия автора курса.

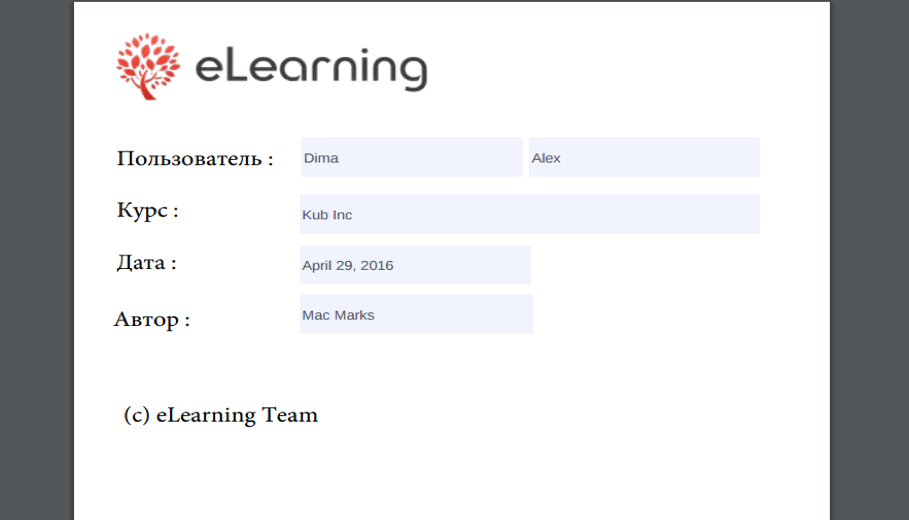


Рисунок 6.14 – Сертификат об успешном прохождении курса