

ОЛИМПИАДА ШКОЛЬНИКОВ «ШАГ В БУДУЩЕЕ»

**НАУЧНО-ОБРАЗОВАТЕЛЬНОЕ СОРЕВНОВАНИЕ
«ШАГ В БУДУЩЕЕ, МОСКВА»**

629

регистрационный номер

Информатика и системы управления(ИУ)

название факультета

Информационные системы и коммуникации(ИУЗ)

название кафедры

Разработка информационно-обучающей смарт-платформы для изучения

программирования

название работы

Автор:

Вейс Иван Анатольевич

фамилия, имя, отчество

ГБОУ школа №1298, 11”Т”

наименование учебного заведения, класс

Научный руководитель:

Павлов Юрий Николаевич

фамилия, имя, отчество

МГТУ им. Баумана

место работы

Доктор технических наук

звание, должность

подпись научного руководителя

Москва - 2019

Аннотация

В проекте рассмотрены проблемы низкой эффективности и доступности изучения языков программирования. Основная цель проекта - значительно улучшить качество уроков программирования в любых образовательных учреждениях, а также упростить и сделать доступнее изучение этого предмета для любого начинающего в этой сфере. Цель моей работы заключается в создании онлайн-платформы, обучающей языкам программирования.

Платформа представляется в виде сайта, включающего в себя универсальную и доступную для любого уровня обучающегося систему. Платформа содержит: теорию, практические задания, свободную компиляцию, контроль учителя за выполнением заданий учениками, а также другие полезные вспомогательные функции. Сайт находится в общей доступности, поэтому каждый желающий может начать изучать программирование в любую минуту с любого современного устройства.

Главное преимущество моего проекта – возможность выполнять поэтапные задания, определяя верность написания программы, видеть необходимый теоретический материал и дальше планомерно изучать его. Опросив учеников моей школы и сравнив скорость работы моей платформы и pascalABC.NET, я выявил, что мой способ обучения намного удобнее и эффективнее, а также имеет множество технических плюсов. Удаленная компиляция в разы быстрее работы программ на устройствах школы, не требует установки, не зависит от мощности компьютера.

Содержание

Введение	4
1 Аналитическая часть	5
1.1 Цели и задачи	5
1.2 Актуальность	6
1.2.1 Структура	6
1.2.2 Практика	6
1.2.3 Удобное оформление	7
1.2.4 Статистика эффективности	8
1.2.5 Ведение уроков	8
2 Техническая часть	9
2.1 Преимущества	9
2.2 Устройство платформы	10
2.3 Дополнительные возможности	11
3 Технологическая часть	12
Заключение	13
Ссылки	14

Введение

В настоящее время информационные технологии играют ключевую роль в нашей жизни: глобально распространен интернет, используется бесчисленное множество разнообразных систем для любых операций с данными. Современный человек обязан уметь пользоваться этими технологиями и представлять, как всё устроено. Для обслуживания различных систем и создания новых требуется владение определёнными навыками, в особенности востребовано знание языков программирования. Число рабочих мест по статистике headhunter за последние 5 лет в этой сфере увеличилось в 5 раз.

На сегодняшний день существует явная нехватка программистов, которая тесно связана с недостаточной эффективностью уроков информатики в школе. Столкнувшись с указанной проблемой лично, я увидел пути её решения и составил данный проект. Этот проект мог бы стать отличной альтернативой школьного курса программирования или просто дать возможность любому желающему получить базовый уровень знаний pascal, python или другого языка.

На уроках информатики учителю приходится длительно объяснять назначение команд языка программирования и чаще всего теория оторвана от практики, хотя ученику намного проще и эффективнее будет прочитать ту же информацию и сразу закрепить новое знание на практике. Кроме того, в школах компьютеры часто бывают с разными ПО, или устаревшие, или невозможно что-либо установить на них по другим причинам, поэтому с практикой на них возникают проблемы. Тогда в качестве решения наиболее распространённых проблем была предложена идея платформы, которая бы хранила теорию и задачи для тренировки и могла бы отображать результаты ученика на компьютере учителя. Решением задачи стал обучающий сайт, поддерживающий все эти функции. Такой вариант значительно упрощает работу учителя и ученика, а также повышает эффективность изучения.

1 Аналитическая часть

1.1 Цели и задачи

Цели проекта:

- Улучшить организацию учебного процесса и повысить его эффективность;
- Создать онлайн платформу, позволяющую изучать программирование быстрее и продуктивнее;
- Сделать изучение языков программирования доступным для любого желающего.

Задачи проекта:

- Создать структуру сайта;
- Создать систему быстрого перехода между страницами сайта;
- Привязать к сайту удаленный компилятор free pascal (план минимума);
- Создать личный кабинет для разделения авторизированных и незарегистрированных пользователей с отображением рейтинга ученика;
- Создать систему регистрации и авторизации с соответствующими полями;
- Создать и настроить рейтинговую систему;
- Создать систему сохранения данных прогресса изучения;
- Создать систему отображения результатов учеников на устройстве учителя;
- Составить теорию по этапам;
- Составить задачи по темам;
- Создать онлайн проверку на корректность кода;
- Добавить прочие компиляторы (план максимума);
- Создать систему динамической генерации файлов с программами пользователей и их запросами ввода;

- Создать систему считывания ip адреса клиента в зависимости от состояния маски браузера;
- Создать систему считывания пользователя в сети по его индивидуальному кешу(набор символов, зашифрованный через md5), ip и COOKIE.

1.2 Актуальность

1.2.1 Структура

Большим плюсом в любой методике обучения является систематизация теории: таблицы и типовые примеры программ значительно упрощают изучение материала. См. рис. 1.

тип	принимаемые значения
integer	целые числа
real	вещественные числа
boolean	логические значения True и False
string	строка из любых символов
char	один любой символ
const	неизменяемое число

Пример присвоения типов переменных

```
var
a,b: integer;
d: real;
text: string;
const
c = 34;
```

Рисунок 1

1.2.2 Практика

Важную роль играет непосредственность применения полученного знания:

Платформа предоставляет возможность решать задачи по изучаемым темам онлайн, чтобы закреплять пройденный материал. Результаты учеников записываются и отображаются в системе учителя, а это значит, что он может наблюдать как за классной, так и домашней работой, а затем быстро проверять результаты.

Результаты работы программы на сайте и на компьютере (через pascalABC.NET и др.) не имеют различий, поэтому можно всегда быть уверенным, что ваш код актуален для других компиляторов. См. рис. 2.

ЗАДАЧИ ПО ГЛАВЕ 1

Задача №1
Ввод/вывод

Напишите программу, которая принимает число, после выводит надпись "Вам [это число] лет".

Пример:

Ввод	Вывод
20	Вам 20 лет

```
program test_in_out;  
var a: integer;  
begin  
  readln(a);  
  writeln("Вам ",a," лет");  
end.
```

Вывод программы	Вывод компилятора
	test.txt Linking ./pascal/test 10 lines compiled, 0.1 sec
Ввод данных(строк: 1)	
50	
<input type="button" value="Ввести"/>	
<input type="button" value="Компилировать"/>	

Рисунок 2

1.2.3 Удобное оформление

Для перехода между главами и темами создана анимация, подтверждающая сделанный выбор, что позволяет проще передвигаться по сайту. Разработанная на CSS и JQUERY анимация работает как на смартфонах, так и компьютерах. На компьютерах стоит направить мышку на иконку главы и ползунок сам переместится на неё в то время, как на телефонах сохраняется возможность нажатия на экран с мгновенным переходом.

1.2.4 Статистика эффективности

По данным исследования подход объединения теории и практики может повысить эффективность обучения на 30 % (запоминаемый материал увеличивается на 30 процентов при возможности сразу реализовать метод программирования на практике). Плюс ко всему, видимость результатов вовлекает учеников в рейтинговую систему, что в итоге повышает мотивацию к обучению ещё на 15 процентов.

А также ученику или учителю не придется:

- искать разрозненные курсы (всё, что надо, поэтапно дано на сайте), 5 %;
- искать отдельные задачи для самостоятельного решения, 5%;
- устанавливать компилятор и другие вспомогательные программы;

Итоговая эффективность обучения повышается более, чем на 50%.

1.2.5 Ведение уроков

Учителю не придётся по несколько раз объяснять назначение каждой команды языка программирования, ученик получает ту же информацию на платформе, если есть необходимость – прочитывает вновь, пишет программу для закрепления пройденной темы, и, убедившись в знании материала, изучает дальше. Такой способ позволяет ученику изучать теорию индивидуально.

Завершив один урок, можно моментально вернуться к пройденному материалу или перейти на следующую ступень изучения. В качестве домашней работы можно задавать определенный набор задач из задачника или задания по главам. Учителю остаётся лишь только оценить результаты выполнения выбранных им задач, не проверяя (как это обычно происходит) каждую работу отдельно. Также считаю, что платформа также полезна для обучающихся, находящихся на дистанционном или очно-заочном обучении. Схема передачи результатов от ученика к учителю проста, см. рис. 3.

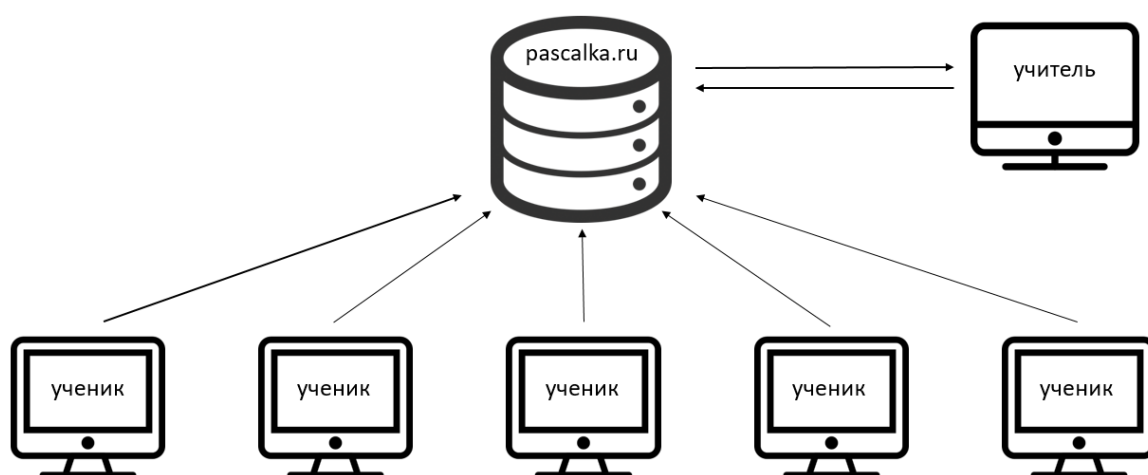


Рисунок 3

2 Техническая часть

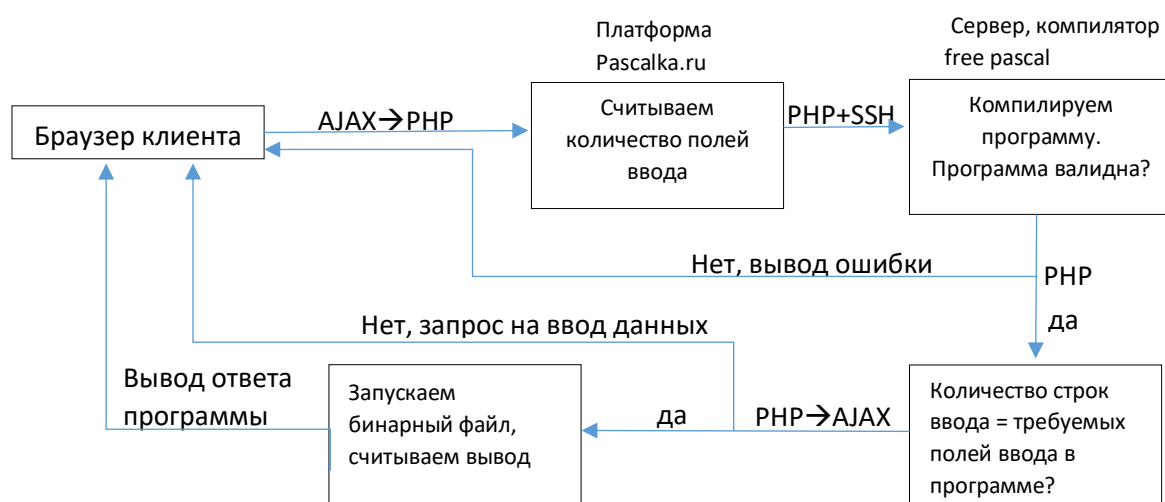
2.1 Преимущества

- Нет необходимости устанавливать приложения;

- Скорость работы программы в разы больше, меньше 0.1с на компиляцию 10ти строк, 0.5с на обработку запроса(pascalABC.NET сработал за 4.5с);
- Поддержка любого ПО;
- Поставлена задача с примерами ввода/вывода;
- Есть проверка на верность работы программы;
- Есть облачное хранение программ
- Есть удалённая видимость учителя результата написания программы учеником

2.2 Устройство платформы

Сложнейшей частью структуры сайта является процесс работы программ на разных языках программирования, так выглядит процесс на примере языка pascal в упрощённом виде:



Все запросы к серверу осуществляются через технологию ajax, то есть без перезагрузки страницы. При переходе на другую тему(страницу с теорией) меняется лишь запрашиваемый блок, а не вся страница, включая кеш. Для этого я создал систему, которая по клику скачивает через ajax с сервера требуемый контент и замещает ранее открытый блок. Такой механизм в разы экономит время работы платформы, минимизирует требования устройства клиента и не позволяет перегружать сервер.

Платформа подключена к базе данных, в которой хранится зашифрованная информация о пользователях. Существует таблица всех зарегистрированных пользователей и динамические таблицы на каждого клиента, где хранятся результаты выполнения задач, определяющие рейтинг человека.

2.3 Дополнительные возможности

Также созданы дополнительные функции, облегчающие написание программы, например, известность количества строк на ввод; вывод номера строки, в которой была встречена ошибка. Как и на любой другой платформе, у меня существует личный кабинет для идентификации личности, в нём отображается прогресс изучения языка по итогам выполнения заданий.

У авторизованного пользователя программы сохраняются и, если тот зайдёт на платформу вновь, будут отображаться в полях выполненных задач. Сохранение и изменение программ таких клиентов происходит динамически с помощью собственной системы на сервере. Незарегистрированные пользователи также смогут практиковаться, однако без сохранения результатов.

3 Технологическая часть

Весь код программы платформы написан вручную, без использования каких-либо CMS. Из этого следует, что структура кода оптимизирована и не содержит лишних функций или переменных, которые могли бы замедлять работу сайта. Помимо jquery(библиотеки js) и recaptcha(анти-бот от google) не было использовано никаких дополнительных плагинов, библиотек и шаблонов.

В ходе выполнения проекта было закреплено и пройдено множество функций и операторов на языках: HTML, CSS, JS+JQUERY(много практики создания запросов на ajax), PHP, а также немного на языке оболочек на Linux, Pascal, Python и C++.

Отдельное внимание я уделил технологии AJAX, позволяющей без перезагрузки страницы отправлять данные серверу через отдельной запрос формата XMLHttpRequest с типом данных json. Для этого были рассмотрены принципы осуществления асинхронной отправки нескольких запросов, формат передаваемых данных, методы отправки. Для создания онлайн компиляции на удаленном (втором) сервере я изучил способы подключения к серверу (как непосредственно с главного сервера через терминал, так и через оболочку ssh2 из файла-обработчика запросов на php), основные команды на linux, способы переопределения каналов ввода и вывода и множество опций компилятора Free Pascal.

Также я узнал, как работать с базами данных на типе SQL, а именно я разобрал весь функционал интернет-версии программы phpmyadmin и научился создавать динамические запросы к серверу на php с помощью функции mysqli/mysql.

Для идентификации пользователей в сети я изучил суперглобальные переменные php и с помощью них создал несколько систем, которые

распределяют файлы в пользовательские папки на временные и постоянные.

Вдобавок была рассмотрена документация по надстройкам конфигураций Apache 2.

Таким образом, была пройдена основная часть веб-программирования, изучены принципы работы с Linux, базами данных на SQL.

Заключение

Делая вывод, можно утверждать, что платформа может быть полезна любому начинающему программисту, она может найти широкое применение в школах или университетах для получения базовых знаний программирования или для самостоятельного изучения. За счёт практических задач материал усваивается быстрее, следовательно, в курс учебной программы можно вложить намного больше теории, чтобы лучше обучить будущих специалистов.

В связи с возможностью выполнения задач на языке pascal в режиме онлайн у платформы нет аналогов в то время, как pascal наиболее всего распространен в школах и используется в части заданий ЕГЭ. Я считаю, что платформа может использоваться в рамках реализации углублённой образовательной программы по предмету “Информатика” в разделе “Программирование”. Также хотелось бы отметить огромные возможности использования созданной платформы для обучения школьников и студентов, находящихся на дистанционном либо очно-заочном обучении.

Ссылки

- 1 Демоверсия проекта: <http://www.pascalka.ru/> ;

Задания

Иван

выйти

Начнем с самого простого!

Скобки < > своей начинкой указывают на то, что должно стоять на их месте. После двойного следа // следует комментарий к коду.

Перейдем к основным правилам написания кода, имеющие место в любой программе: Язык Pascal имеет свой строгий, отличный от других, синтаксис. Однако, как и почти везде, любая строка должна заканчиваться точкой с запятой
Любая программа должна иметь свое имя, поэтому в самом начале пишем:

```
Program <наименование>;
```

Название переменной состоит из любых букв и цифр, причём первый символ обязательно - буква.

Далее начало и конец выполнения программы обуславливается обязательными операторами BEGIN и END.

После тега BEGIN не ставится никаких знаков, в то время как после тега END в конце

- 2 Статистика рабочих мест hh:
<https://stats.hh.ru/#dynamicVacancies%5Bactive%5D=true&dynamic-vacancies%5Bdynamic-vacancies%5D=year>
- 3 Изучение php: [http://php.su/\(http://php.net\)](http://php.su/(http://php.net))
 - 3.1 <http://php.net/manual/ru/book.ssh2.php>
 - 3.2 <http://php.net/manual/ru/function.fopen.php>
 - 3.3 <http://php.net/manual/ru/book.mysql.php>
 - 3.4 <http://php.net/manual/ru/language.variables.superglobals.php>
 - 3.5 <http://php.net/manual/ru/features.remote-files.php>
 - 3.6 <http://php.net/manual/ru/refs.compression.php>
- 4 AJAX <http://api.jquery.com/jquery.ajax/>
- 5 Копирайт текста – источники:
 - 5.1 <https://pas1.ru/>
 - 5.2 <http://nk-inform.narod.ru/pas/pas.htm>

5.3 <http://mif.vspu.ru/books/pascal/types.html>

6 Работа с Linux

6.1 <https://losst.ru/42-komandy-linux-kotorye-vy-dolzhny-znat>

6.2 <https://losst.ru/kak-sozdat-polzovatelya-linux>

6.3 [https://wiki.archlinux.org/index.php/Environment_variables_\(%D0%A0%D1%83%D1%81%D1%81%D0%BA%D0%B8%D0%B9\)](https://wiki.archlinux.org/index.php/Environment_variables_(%D0%A0%D1%83%D1%81%D1%81%D0%BA%D0%B8%D0%B9))

7 Настройки Apache 2

7.1 <https://losst.ru/ustanovka-i-nastrojka-servera-apache>

7.2 <https://help.ubuntu.ru/wiki/apache2>

7.3 http://www.codenet.ru/webmast/apache/Apache-WindowsNT/apache_config.php