# **Научно исследовательская часть**

## **2.1 Постановка задачи проектирования**

Когда речь заходит о покупке чего-либо в наше время, покупатель захочет сэкономить время и силы, посмотрев интересующий товар на сайте магазина, нежели самому прийти в этот магазин и там часами пытаться выбрать желаемое. При необходимости купить попугая или клетку для него, покупателю необходима возможность заранее посмотреть то, что его интересует, прочитать подробную информацию об этом. Например, выбрать попугая, в зависимости от средней продолжительности жизни его вида, выбрать подходящего поставщика попугая и многое другое. Кроме того, после успешного выбора, было бы неплохо «забронировать» выбранное, чтобы покупатель смог показать свои намерения совершить покупку, а также чтобы не было путаницы с другими покупателями, которые захотят купить тоже самое. Удобнее и быстрее всего сделать это, будет с помощью соответствующего веб-приложения. Поэтому необходимо реализовать такое веб-приложение, с помощью которого покупатель смог бы заранее посмотреть интересующее, а также в случае успешного выбора, смог бы и забронировать товар.

Для того, чтобы реализовать наиболее эффективное и удобное веб-приложение, необходимо использовать базу данных MySql, язык гипертекстовой разметки HTML, каскадные таблицы стилей CSS, язык программирования PHP.

**2.2 Описание предметной области.**

Веб-приложение — клиент-серверное приложение, в котором клиент взаимодействует с веб-сервером при помощи браузера. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются межплатформенными службами.

Веб-приложения стали широко использоваться в конце 1990-х — начале 2000-х годов.

Существенное преимущество построения веб-приложений для поддержки стандартных функций браузера заключается в том, что функции должны выполняться независимо от операционной системы данного клиента. Вместо того, чтобы писать различные версии для Microsoft Windows, Mac OS X, GNU/Linux и других операционных систем, приложение создаётся один раз для произвольно выбранной платформы и на ней разворачивается. Однако различная реализация, CSS, DOM и других спецификаций в браузерах может вызвать проблемы при разработке веб-приложений и последующей поддержке. Кроме того, возможность пользователя настраивать многие параметры браузера (например, размер шрифта, цвета, отключение поддержки сценариев) может препятствовать корректной работе приложения.

Другой (менее универсальный) подход заключается в использовании Adobe Flash, Silverlight или Java-апплетов для полной или частичной реализации пользовательского интерфейса. Поскольку большинство браузеров поддерживает эти технологии (как правило, с помощью плагинов), Flash- или Java-приложения могут выполняться с легкостью. Так как они предоставляют программисту больший контроль над интерфейсом, они способны обходить многие несовместимости в конфигурациях браузеров, хотя несовместимость между Java- или Flash-реализациями на стороне клиента может приводить к различным осложнениям.

На 2015 год технологию Adobe Flash не поддерживают Chrome, Safari, и другие популярные браузеры.

В связи с архитектурным сходством с традиционными клиент-серверными приложениями, в некотором роде «толстыми» клиентами, существуют споры относительно корректности отнесения подобных систем к веб-приложениям; альтернативный термин «полнофункциональное приложение интернета» (англ. Rich Internet Applications).

Используются подходы к архитектурному проектированию веб - приложений. Веб-приложение состоит из клиентской и серверной частей, тем самым реализуя технологию «клиент-сервер».

Клиентская часть реализует пользовательский интерфейс, формирует запросы к серверу и обрабатывает ответы от него.

Серверная часть получает запрос от клиента, выполняет вычисления, после этого формирует веб-страницу и отправляет её клиенту по сети с использованием протокола HTTP.

Само веб-приложение может выступать в качестве клиента других служб, например, базы данных или другого веб-приложения, расположенного на другом сервере. Ярким примером веб-приложения является система управления содержимым статей Википедии: множество её участников могут принимать участие в создании сетевой энциклопедии, используя для этого браузеры своих операционных систем (будь то Microsoft Windows, GNU/Linux или любая другая операционная система) и не загружая дополнительных исполняемых модулей для работы с базой данных статей.

В настоящее время набирает популярность новый подход к разработке веб-приложений, называемый Ajax. При использовании Ajax страницы веб-приложения не перезагружаются целиком, а лишь догружают необходимые данные с сервера, что делает их более интерактивными и производительными.

Также в последнее время набирает большую популярность технология WebSocket, которая не требует постоянных запросов от клиента к серверу, а создает двунаправленное соединение, при котором сервер может отправлять данные клиенту без запроса от последнего. Таким образом появляется возможность динамически управлять контентом в режиме реального времени.

Для создания веб-приложений на стороне сервера используются разнообразные технологии и любые языки программирования, способные осуществлять вывод в стандартную консоль.

На стороне клиента используется:

* Для реализации GUI
* HTML, XHTML
* CSS
* Для формирования запросов, создания интерактивного и независимого от браузера интерфейса:
* ActiveX
* Adobe Flash, Adobe Flex
* Java
* JavaScript
* Silverlight

**2.3 Анализ аналогов и прототипов**

Для того, чтобы сделать хороший проект, необходимо проанализировать и сравнить уже существующие, самые часто используемые веб-приложения для выбора попугаев, и выделить их основные преимущества и недостатки. Воспользовавшись недочётами других разработчиков, не повторим их в своем проекте и сделаем более удобное и правильное веб-приложение.

Посмотрим на преимущества и недостатки веб-приложения https://zoostore.pro

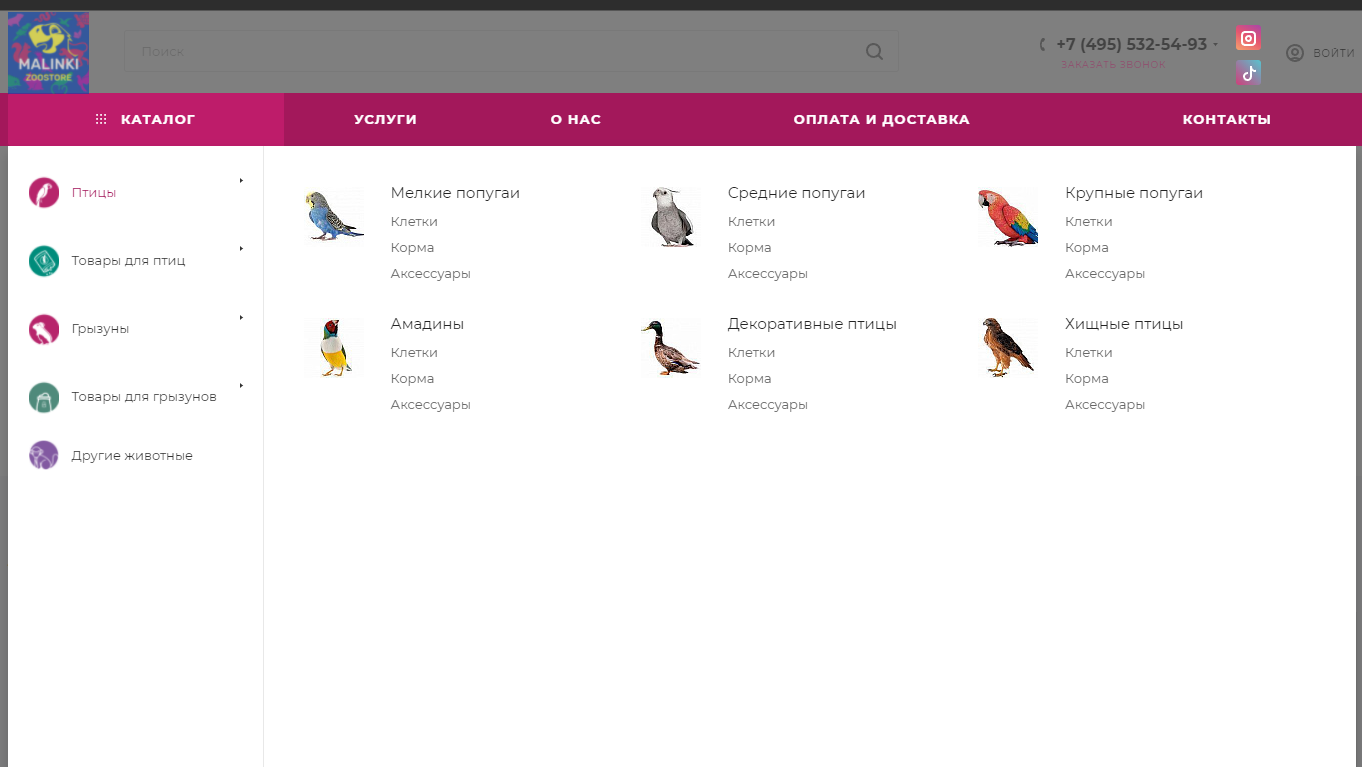
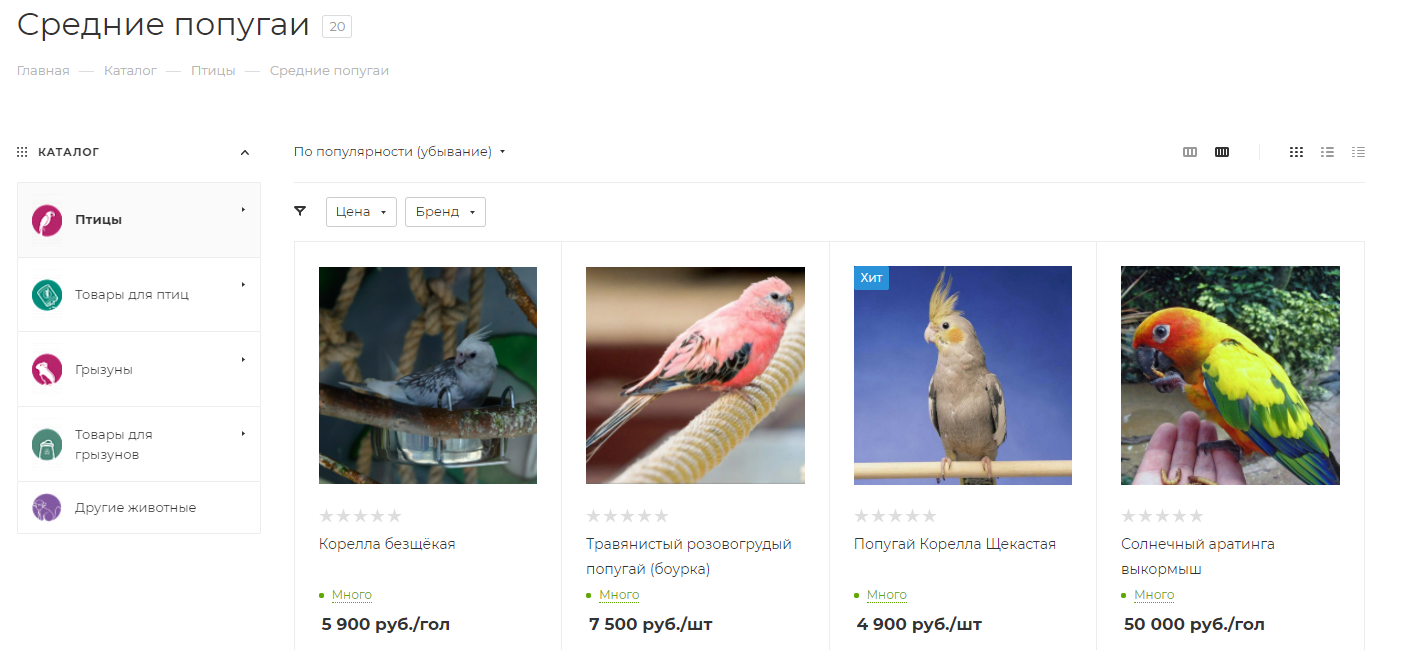


Рисунок 1.1 - Интерфейс сайта https://zoostore.pro/

Преимущества:

* простой и понятный интерфейс;
* возможность выбора попугая по размерам;
* фотографии попугаев, с помощью них можно понять, как выглядит попугай и выбрать понравившегося;
* отображение наличия попугаев;

Недостатки:

* отсутствие возможности выбора попугаев исходя из предпочитаемого вида (на рисунке 1.1 можно увидеть, что выбрать можно только размеры попугаев).

Далее проанализируем сайт https://happybirds.ru/.

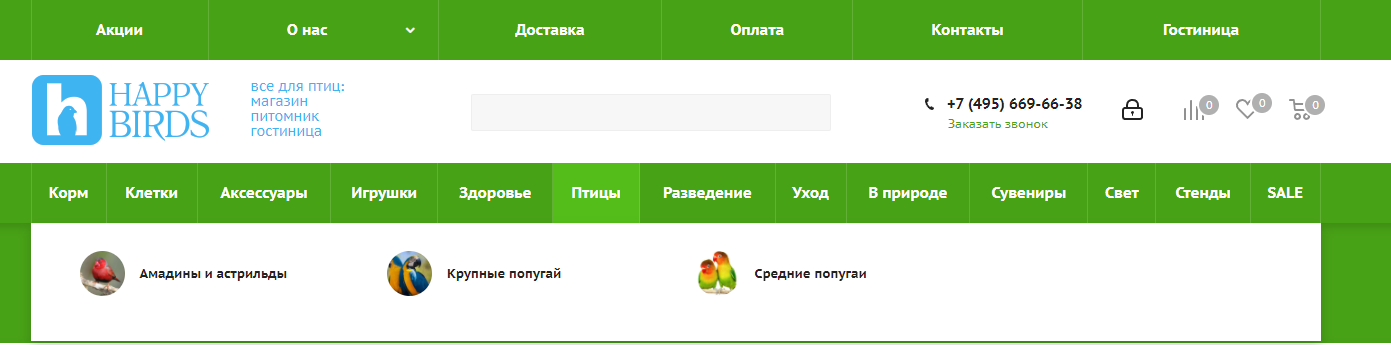
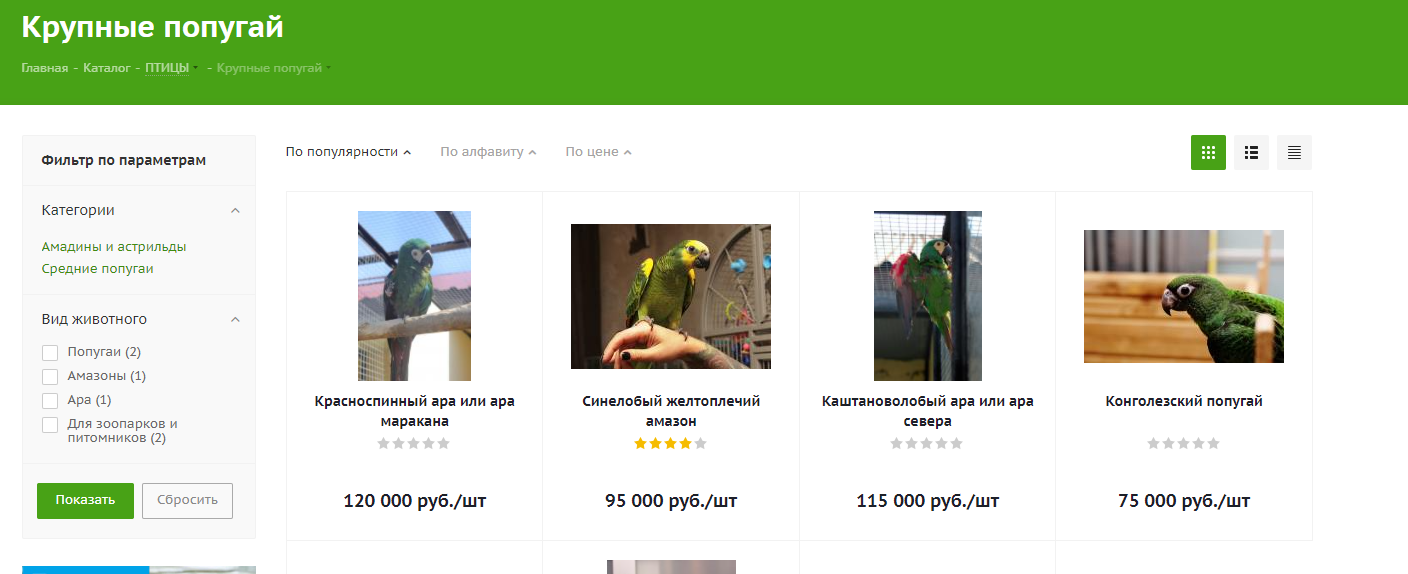


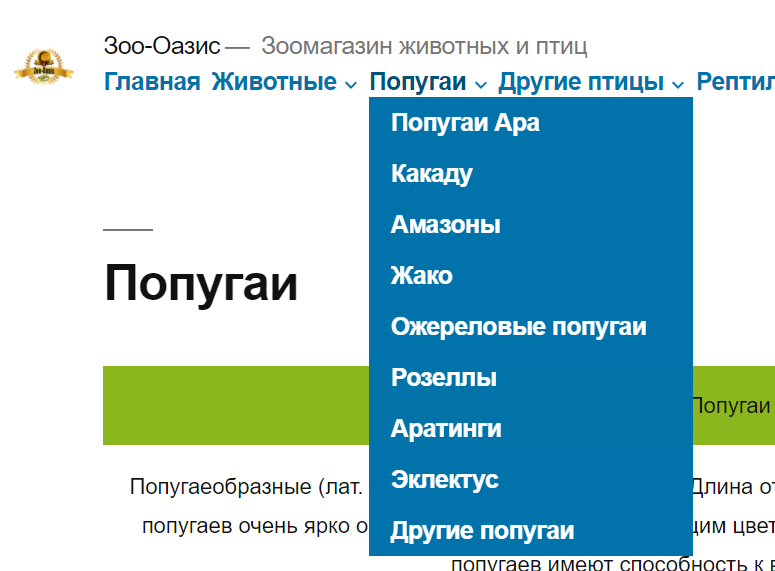
Рисунок 1.2 – Интерфейс сайта https://happybirds.ru/.

Преимущества:

* приятный и удобный интерфейс;
* фотографии попугаев, с помощью них можно понять, как выглядит попугай и выбрать понравившегося;

Недостатки:

* отсутствие возможности выбора попугаев исходя из предпочитаемого вида;
* отсутствие отображения наличия попугаев.

Сайт <https://zoo-oasis.ru/popugai/>.

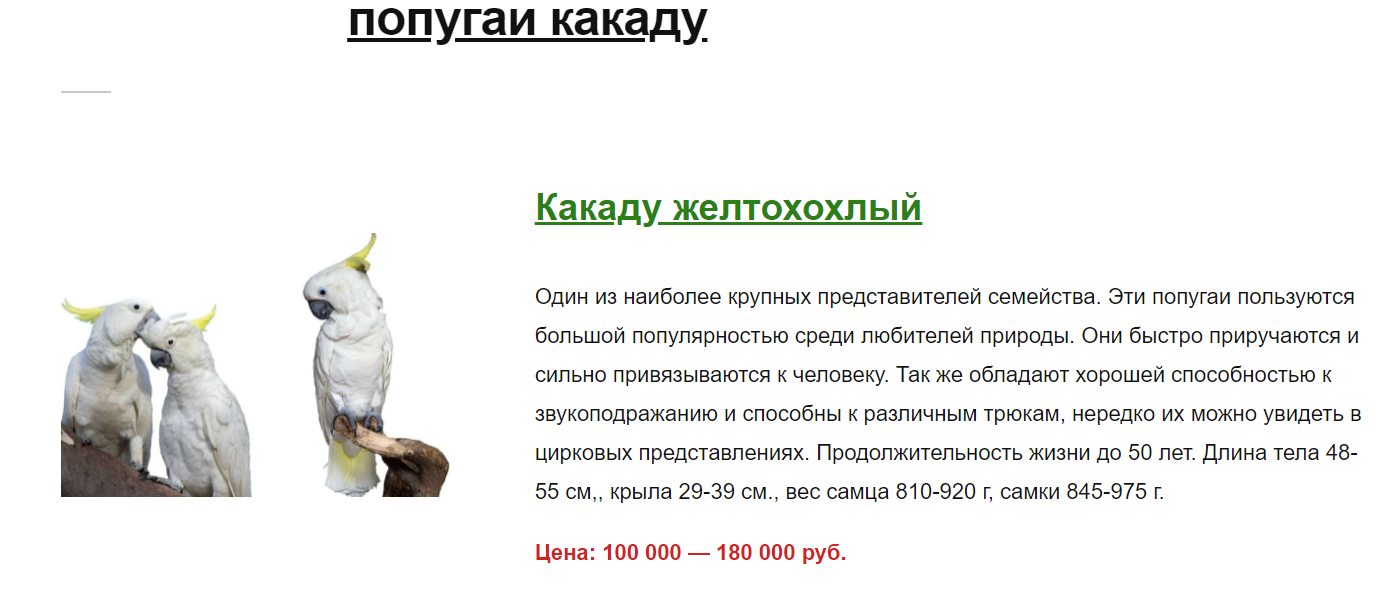


Рисунок 1.4 - Интерфейс сайта <https://zoo-oasis.ru/popugai/>

Преимущества:

* возможность выбора попугаев по их виду;
* подробное описание попугаев (размеры, вес, продолжительность жизни);

Недостатки:

* скудный и некрасивый интерфейс;
* невозможность заказать, забронировать, купить попугаев (только лишь просмотр).

Проанализировав аналоги и прототипы, можно сделать вывод, что существующие веб-приложения не позволяют удобно и быстро решать поставленные задачи, поэтому необходимо разработать собственное веб-приложение, которое бы предоставляло удобную возможность просмотра попугаев и клеток для них, а также возможность их дальнейшего бронирования.

## **2.4 Перечень задач, подлежащих решению в процессе разработки.**

* разработка пользовательского интерфейса с использованием HTML и CSS;
* разработка базового функционала веб-приложения;
* разработка запросов и переходов по страницам сайта;
* привязка запросов и переходов по страницам к пользовательскому интерфейсу;
* привязка базы данных к веб-приложению для хранения данных;
* тестирование разработанного приложения;
* исправление выявленных ошибок.

**2.5 Обоснование выбора инструментов и платформы для разработки.**

Для успешной разработки проекта, используется операционная система Windows 10. Windows 10 — [операционная система](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) для [персональных компьютеров](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D1%81%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80) и [рабочих станций](https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D0%B1%D0%BE%D1%87%D0%B0%D1%8F_%D1%81%D1%82%D0%B0%D0%BD%D1%86%D0%B8%D1%8F), разработанная корпорацией [Microsoft](https://ru.wikipedia.org/wiki/Microsoft" \o "Microsoft) в рамках семейства [Windows NT](https://ru.wikipedia.org/wiki/Windows_NT).

Данная платформа предоставляет широкий набор инструментов для работы с ОС Windows, например, язык программирования Си с использованием базовых функций интерфейсов программирования приложений операционных систем Microsoft Windows – Windows API. WinApi – это набор функций, работающих под управлением ОС Windows. С помощью WinAPI можно создавать различные оконные процедуры, диалоговые окна, программы и даже игры, а также работать с операционной системой на прямую.

Согласно Statcounter.com, общий рейтинг операционных систем, показывает, что лидером является Windows, который установлен на 76,58% всех компьютеров, OS X — 18,93% и Linux — 1,62%. Платформа Windows является самой популярной и доступной, также для этой платформы создано много удобных средств, для облегчения выполнения проекта.

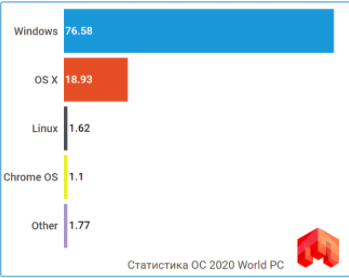


Рисунок 1.6 - Статистика использования операционных систем в мире.

В качестве среды разработки используется Visual Studio Code. Интегрированная среда разработки Visual Studio Code — это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Интегрированная среда разработки (IDE) представляет собой многофункциональную программу, которую можно использовать для различных аспектов разработки программного обеспечения. Помимо стандартного редактора и отладчика, которые существуют в большинстве сред IDE, Visual Studio Code включает в себя компиляторы, средства автозавершения кода, графические конструкторы и многие другие функции для упрощения процесса разработки.

Ниже перечислены некоторые популярные возможности Visual Studio Code, которые помогут вам повысить продуктивность разработки программного обеспечения.

* волнистые линии и [быстрые действия](https://docs.microsoft.com/ru-ru/visualstudio/ide/quick-actions?view=vs-2019);

Волнистые линии обозначают ошибки или потенциальные проблемы кода прямо во время ввода. Эти визуальные подсказки позволяют устранять проблемы немедленно и не ждать, пока ошибка будет обнаружена во время сборки или запуска программы. Если навести указатель мыши на волнистую линию, на экран будут выведены дополнительные сведения об ошибке. Кроме того, в поле слева может появляться значок лампочки с быстрыми действиями по устранению ошибки.

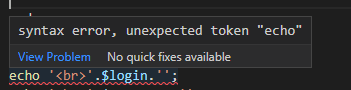


Рисунок 1.7. Волнистые линии и быстрые действия

* поиск в Visual Studio Code

Среда Visual Studio Code может показаться сложной, ведь там столько разных меню, параметров и свойств. Чтобы быстро находить функции интегрированной среды разработки и элементы кода, в Visual Studio Code представлен единый компонент поиска (**CTRL**+**Q**).

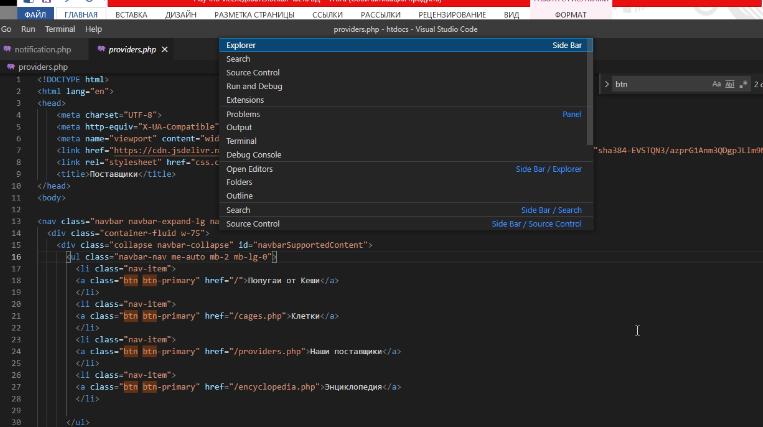


Рисунок 1.8. Поиск в Visual Studio Code

Согласно опросу, проведенного среди 3240 разработчиков в феврале 2018 организацией Standard, разработчики выделили такие средства разработки, как наиболее часто используемые:

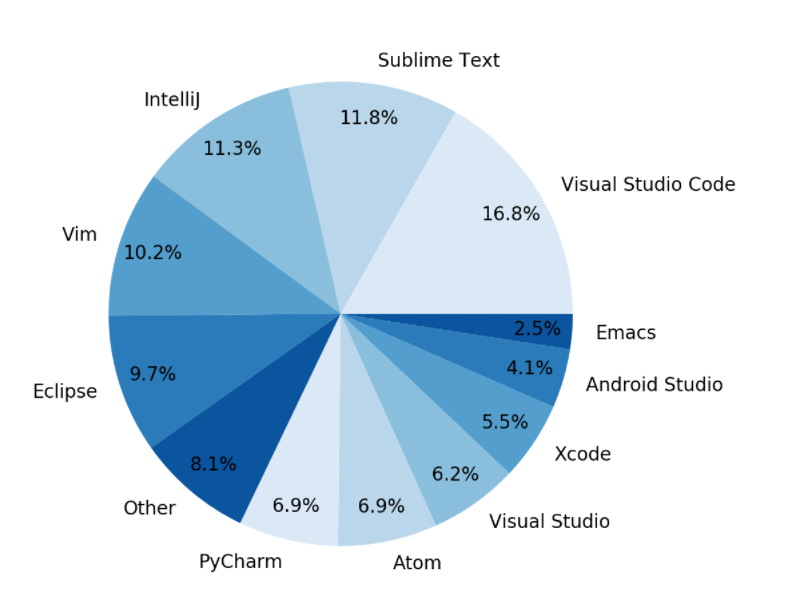


Рисунок 1.9 - Статистика использования платформ для разработки систем в мире.

Первое место среди IDE у Visual Studio Code с 16.8% использующих его разработчиков.

Также, Visual Studio Code обладает хорошим отладчиком, с помощью которого можно легко отследить ошибки в коде и без проблем их исправить, позволяет удобно и быстро создавать проекты и работать с ними.

По сравнению со средой разработки Eclipse, Visual Studio Code имеет более простой интерфейс в использовании.

Для разработки проекта можно использовать следующие инструменты:

* язык гипертекстовой разметки HTML;

HTML (от англ. HyperText Markup Language — «язык гипертекстовой разметки») — стандартизированный язык разметки документов для просмотра веб-страниц в браузере. Веб-браузеры получают HTML документ от сервера по протоколам HTTP/HTTPS или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора.

Элементы HTML являются строительными блоками HTML страниц. С помощью HTML разные конструкции, изображения и другие объекты такие как интеракивная веб-формы могут быть встроены в отображаемую страницу. HTML предоставляет средства для создания заголовков, абзацев, списков, ссылок, цитат и других элементов. Элементы HTML выделяются тегами, записанными с использованием угловых скобок. Такие теги, как <img /> и <input />, напрямую вводят контент на страницу. Другие теги, такие как <p>, окружают и оформляют текст внутри себя и могут включать другие теги в качестве подэлементов. Браузеры не отображают HTML-теги, но используют их для интерпретации содержимого страницы.

В HTML можно встроить программый код на языке программирования JavaScript, для управления поведением и содержанием веб-страниц. Также включение CSS в HTML описывает внешний вид и макет страницы.

* каскадные таблицы стилей CSS;

CSS используется создателями веб-страниц для задания цветов, шрифтов, стилей, расположения отдельных блоков и других аспектов представления внешнего вида этих веб-страниц. Основной целью разработки CSS являлось отделение описания логической структуры веб-страницы (которое производится с помощью HTML или других языков разметки) от описания внешнего вида этой веб-страницы (которое теперь производится с помощью формального языка CSS). Такое разделение может увеличить доступность документа, предоставить большую гибкость и возможность управления его представлением, а также уменьшить сложность и повторяемость в структурном содержимом.

Кроме того, CSS позволяет представлять один и тот же документ в различных стилях или методах вывода, таких как экранное представление, печатное представление, чтение голосом (специальным голосовым браузером или программой чтения с экрана) или при выводе устройствами, использующими шрифт Брайля.

* язык программирования PHP;

PHP (англ. PHP: Hypertext Preprocessor — «PHP: препроцессор гипертекста»; первоначально PHP/FI (Personal Home Page / Form Interpreter), а позже названный Personal Home Page Tools — «Инструменты для создания персональных веб-страниц») — скриптовый языкобщего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов.

Язык и его интерпретатор (Zend Engine) разрабатываются группой энтузиастов в рамках проекта с открытым кодом. Проект распространяется под собственной лицензией, несовместимой с GNU GPL.

В области веб-программирования, в частности серверной части, PHP — один из популярных сценарных языков (наряду с JSP, Perl и языками, используемыми в ASP.NET).

Популярность в области построения веб-сайтов определяется наличием большого набора встроенных средств и дополнительных модулей для разработки веб-приложений. Основные из них:

* автоматическое извлечение POST- и GET-параметров, а также переменных окружения веб-сервера в предопределённые массивы;
* взаимодействие с большим количеством различных систем управления базами данных через дополнительные модули (MySQL, MySQLi, SQLite, PostgreSQL, Oracle Database (OCI8), Microsoft SQL Server, Sybase, ODBC, mSQL, IBM DB2, Cloudscape и Apache Derby, Informix, Ovrimos SQL, Lotus Notes, DB++, DBM, dBase, DBX, FrontBase, FilePro, Ingres II, SESAM, Firebird и InterBase, Paradox File Access, MaxDB, интерфейс PDO, Redis);
* автоматизированная отправка HTTP-заголовков;
* работа с HTTP-авторизацией;
* работа с cookies и сессиями;
* работа с локальными и удалёнными файлами, сокетами;
* обработка файлов, загружаемых на сервер;
* работа с XForms.

По состоянию на 2010-е годы используется сотнями тысяч разработчиков; согласно рейтингу корпорации TIOBE, базирующемуся на данных поисковых систем, в мае 2016 года PHP находился на 6 месте среди языков программирования.

Входит в LAMP — распространённый набор программного обеспечения для создания и хостинга веб-сайтов (Linux, Apache, MySQL, PHP).

Среди сайтов, использующих PHP — Facebook, Wikipedia, Yahoo!, Baidu.

* база данных MySql;

Первое место среди IDE у Visual Studio Code с 16.8% использующих его разработчиков.

MySQL — свободная реляционная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация Oracle, получившая права на торговую марку вместе с поглощённой Sun Microsystems, которая ранее приобрела шведскую компанию MySQL AB. Продукт распространяется как под GNU General Public License, так и под собственной коммерческой лицензией. Помимо этого, разработчики создают функциональность по заказу лицензионных пользователей. Именно благодаря такому заказу почти в самых ранних версиях появился механизм репликации.

MySQL является решением для малых и средних приложений. Входит в состав серверов WAMP, AppServ, LAMP и в портативные сборки серверов Денвер, XAMPP, VertrigoServ. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей. Более того, СУБД MySQL поставляется со специальным типом таблиц EXAMPLE, демонстрирующим принципы создания новых типов таблиц. Благодаря открытой архитектуре и GPL-лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц.

MySQL имеет API и коннекторы для языков Delphi, C, C++, Эйфель, Java, Лисп, Perl, PHP, Python, Ruby, Smalltalk, Компонентный Паскаль, Tcl и Lua, библиотеки для языков платформы .NET.