**Основные определения ПО**

**Програ́ммное обеспе́чение** (**ПО**) — программа или множество программ, используемых для управления компьютером.

Имеются и другие определения :

1. совокупность программ системы обработки информации и программных документов, необходимых для эксплуатации этих программ
2. компьютерные программы, процедуры и, возможно, соответствующая документация и данные, относящиеся к функционированию компьютерной системы.

Программное обеспечение является одним из видов обеспечения вычислительной системы, наряду с техническим (аппаратным), математическим,

информационным, организационным, методическим и правовым обеспечением.

**Виды классификации ПО**

**Программное обеспечение делится на 3 категории:**

**1)**прикладное;  
**2)**системное;  
**3)**инструментальное;

**Системное** — программа, предназначенная для эксплуатации и технического обслуживания ЭВМ, управления и организации вычислительного процесса при решении любой конкретной

**Прикладное** —  [программа](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0), ориентированная на решение конкретных задач,  рассчитанная на взаимодействие с пользователем. В большинстве [ОС](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) прикладные программы не могут обращаться к ресурсам [компьютера](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80)  напрямую, взаимодействуя с оборудованием и другими программами через ОС.

**Инструментарное –** программа, используемые как инструмент при создании документов в повседневной деятельности.

**Система програмирования**

Система программировани**я** — это система для разработки новых программ на конкретном языке программирования.

Современные системы программирования обычно предоставляют пользователям  мощные и удобные средства разработки программ.  В них входят:

1. Компилятор;
2. Интегрированная среда для разработки;
3. Библиотека стандартны программ;
4. Многооконный режим работы;
5. Встроенный ассемблер;
6. Встроенная справочная служба;

**Этапы подготовки программы**

Этап 1 – Определение проблемы (Перед тем как приступать к кодированию, необходимо четко сформулировать проблему, которую ваша будущая программа должна решать.)

Этап 2–Выработка требований ( Требования вырабатывают для того, чтобы свести к минимуму изменения системы после начала непосредственной разработки. Такие требования должны быть обязательно официальными, т.е. документально оформлены. Так как это гарантирует то, что функциональность системы определяется заказчиком, а не программистом. )

Этап 3 – Создание плана разработки (На данном этапе Вы уже должны в формальном виде составить план разработки программного обеспечения с учётом существующей проблемы и выработанных требований. Иными словами, Вы должны составить план того, как Вы будете действовать дальше.)

Этап 4 – Разработка архитектуры системы (Данный этап также очень важный, так как, не имея хорошей архитектуры, Вы можете решать правильную проблему, но прийти к неправильному решению. Хорошая архитектура программы упрощает программирование, а плохая архитектура усложняет его.)

Этап 5 – Детальное проектирование (На этом этапе проводится проектирование программы на низком уровне, иными словами, здесь проектируются классы и методы, рассматриваются, оцениваются и сравниваются различные варианты и причины выбора окончательных подходов и способов реализации.)

Этап 6 – Кодирование и отладка (После того как код написан, программисту необходимо отладить этот код, чтобы в нем не было никаких ошибок.)

Этап 7 – Тестирование компонентов (После того, как код написан, и проведена отладка, необходимо провести тестирование реализованного функционала.)

Этап 8 – Интеграция компонентов (Когда тестирование всех компонентов закончено, можно переходить к интеграции всех компонентов в единый программный комплекс, этот этап как раз и подразумевает процесс интеграции, т.е. слияния всех компонентов в единую систему.)

Этап 9 – Тестирование всей системы (На данном этапе проводится тестирование всей системы, уже с учётом интеграции всех компонентов. На этом этапе можно выявить проблемы взаимодействия компонентов и устранить их.)

Этап 10 – Сопровождение, внесение изменений и оптимизация. (После запуска программы в промышленную эксплуатацию осуществляется сопровождение этой программы, т.е. внесение изменений на основе выявленных недочетов в процессе эксплуатации системы, а также проводится оптимизация функционала или добавление нового.)

## Языки программирования

Языки высокого уровня похожи на человеческий язык и позволяют программистам сосредоточиться на проблеме.

[Языки программирования низкого уровня](https://programmera.ru/articls/yazyki-programmirovaniya-nizkogo-urovnya/) обращаются непосредственно к «железу», давая ему определенные точные команды, а языки программирования высокого уровня оперируют более абстрактными понятиями.

|  |  |  |
| --- | --- | --- |
|  | Языки высокого уровня | Языки низкого уровня |
|  |  |  |
| Сходства | Имеют общие компиляторы, работающие как для низкоуровневых языков, так и для высокоуровневых | |
| Различия | Далеки от машинных языков | Близки к машинным языкам, содержат набор машинных команд. |
| Понятность и удобность в поиске ошибок | Трудность в понимании и выявлении ошибок |
| Медленные и неэффективные компиляторы | Быстрота и надёжность компиляторов |
| Компактность написанных программ | Написанные программы получаются длинными |

Таблица 1. Языки высокого и низкого уровня

**Операционные системы**

Компьютерная система состоит из нескольких процессоров: оперативной памяти, дисков, клавиатуры, монитора, принтеров, сетевого интерфейса и другого, то есть является отлаженной комплексной системой.

Создание программы, которая следит за компонентами, правильно использует их и при этом выдает стабильную работу, представляет собой крайне сложную задачу. Поэтому компьютеры оснащаются специальным уровнем программного обеспечения, называемого операционной системой.

Операционная система представляет собой комплекс программ, руководящая ресурсами системы, осуществляющая организацию вычислительных процессов и обеспечивающая взаимодействие между пользователями, программистами, прикладными программами, системными приложениями и аппаратным обеспечением компьютера.

Она создана для управления всей работой компьютера, эффективного использования его ресурсов. Простыми слова ОС — это «Большой Босс», который выполняет следующие функции:

1. Управляет свободными ресурсами оперативной памяти, жестких дисков или карт памяти.

2. Руководит загрузкой процессора.

3. Распределяет потоки процессов для создания многозадачности.

4. Устанавливает, обновляет и удаляет программы и приложения.

**Виды операционных систем**

ОС можно классифицировать на Windows и UNIX-подобные. Это два основных вида в которых делаются операционные системы.

Обычно операционные системы разделяют на:

* Многопользовательские — когда может быть несколько пользователей
* Однопользовательские — когда разных пользователей создать нельзя

Типы ОС:

* Графические и текстовые
* Платные и бесплатные
* Открытые и закрытые
* Клиентские и серверные
* Высокой или низкой стабильности к сбоям аппаратного обеспечения
* По разрядности: 16, 32 или 64 битные, раньше были вообще 8 битные
* С высоким или низким уровнем безопасности
* Мобильны или настольные

**Виды ошибок в операционных системах**

Операционная система, являясь комплексом программ может совершать ошибки подобно любой другой программе. Эти ошибки зачастую приводят к неисправности работы компьютера. Давайте же рассмотрим, что оно из себя представляет.

Оши́бка — непреднамеренное, случайное отклонение от правильных действий, поступков, мыслей, разница между ожидаемой или измеренной и реальной величиной.

Джордж А. Миллер в своей работе в 1960 году определил ошибки как все отклонения фактического состояния некоторого объекта от целевого состояния.

Windows:

Самые распространенные ошибки у пользователей OC Windows:

* Ошибка 0x80200056
* Ошибка 0x800F0922
* Ошибка 0x800F0923

Эти ошибки связаны с такими проблемами как: несовместимость ПО, нехватка места на жёстком диске, или остановка процесса обновления. Системные сбои Windows приводят к остановке системы и к «синему экрану» который получил такое название среди пользователей.

* Синий экран — название сообщения о критической системной ошибке (стоп-ошибке) в операционных системах Microsoft Windows.

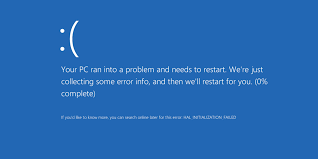


Рисунок 1 ошибка в windows

* Ошибка жесткого диска — является достаточно серьезным сигналом, который свидетельствует о неполадках в работе системы или жесткого диска.

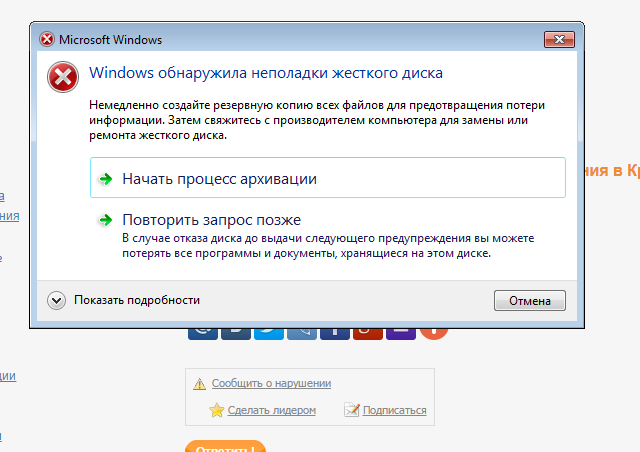


Рисунок 2 ошибка в windows

* Сбой профиля в подавляющем большинстве случае связана с повреждением профиля пользователя или прав на каталог профиля. Это может произойти из-за некорректного завершения работы системы, обновления системы, некорректных изменений в реестре или других сбоев.

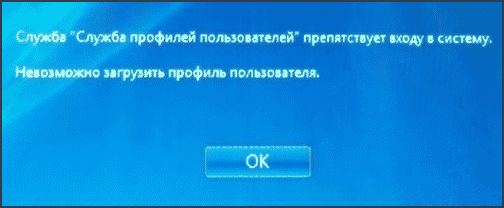


Рисунок 3 ошибка в windows

Linux

* Ubuntu has experienced an internal error — это означает что в вашей системе произошел сбой.

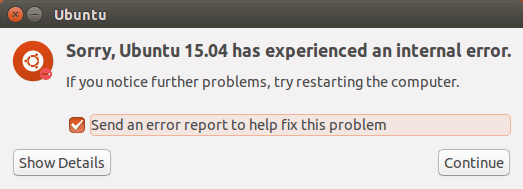


Рисунок 4 ошибка в linex

Mac OS

* Неожиданные перезапуски обычно вызваны установленным программным обеспечением или подключенными к нему устройствами. Если проблема вызывает перезапуск компьютера Mac при каждой попытке запуска, в какой-то момент компьютер Mac может выключиться.

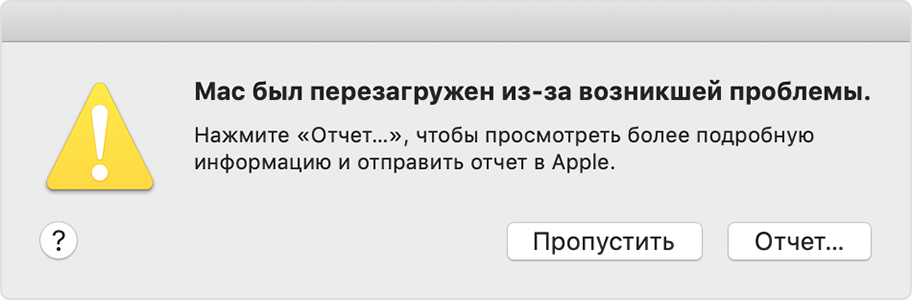


Рисунок 5 ошибка в mac os

**Совместимость операционных систем**

Некоторые программы, устаревшие для современных операционных систем, всё ещё могут пригодиться в наше время для этого, например, в операционной системе Windows существует режим совместимости. Режим совместимости позволяет запускать программное обеспечение, которое исправно работало на предыдущих версиях windows. Например, с помощью режима совместимости на Windows 10 можно запустить программы с Windows XP, Windows Vista, Windows 7 и т. д.

Свойства совместимости имеют такие параметры как:

* Режим совместимости
* Режим пониженной цветности
* Разрешение экрана в 640x480 пикселей
* Выполнение программы от имени администратора

Проведём тест режима совместимости в случайных программах на данном компьютере, возьмём 3 разных программы: браузер Google Chrome, текстовый редактор Word, и компилятор Visual Studio Code. Будем запускать программы в режиме совместимости с Windows 7. Проверка показала, что запуск в режиме совместимости никак не препятствует работе и не вызывает никаких изменений. Так как испытания проводились на Windows 10 и запускаемые программы были новыми и обновлялись с течением времени то они прекрасно работают без режима совместимости, из этого можно сделать вывод что режим совместимости может пригодится только в использовании старых программ, которые давно не обновлялись, или одной и той же программы разных версий.

## Программирование

Задача: создать 3 одинаковые консольные программы на разных языках программирования.

Для выполнения этой задачи понадобится: Python, JavaScript, Pascal. Для одинаковых программ, записанных на разных языках, была выбрана простая задача - вывод суммы двух заданных целых чисел, 2 и 4. Для написания консольных программ используется Visual Studio Code.

На Python задача решалась первой и там она выглядит самой простой так как нету особенных символов, или других усложнений, арифметические знаки и команда вывода выглядят обычно.

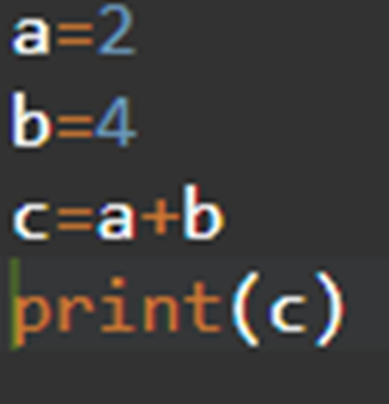


Рисунок 1. решение задачи на Python

Дальше решение на Pascal. Выполнение этой задачи отличается только особенностями языка, а если конкретно то нужно указывать начало(Begin) и конец(End) программы, её название(Program), а также знаки не похожие на обыкновенные арифметические.

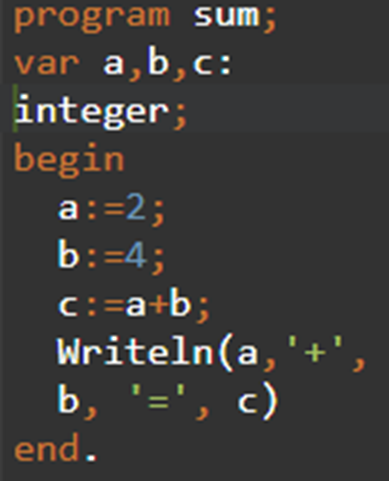


Рисунок 2. решение задачи на Pascal

Для решения этой задачи на JavaScript потребуется создать html страницу чтобы вывести результат написанных команд в консоли веб-страницы.

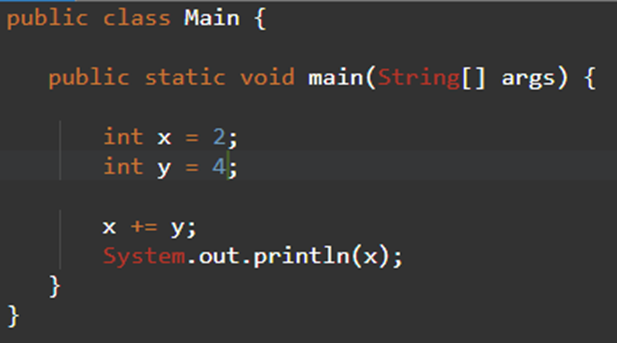


Рисунок 3. решение задачи на JavaScriptВыполнение задачи на этих трёх языках никак не изменило получаемый результат, но изменило подход к решению задачи, учитывая особенности конкретного языка.

**№3 Маркетинг**

В рыночных условиях успешная деятельность предприятий и фирм во многом зависит от качества умелой и грамотной маркетинговой деятельности. В связи с этим возрастает роль профессиональной подготовки специалистов, обладающих умением исследовать и выявлять потребности рынка, разрабатывать стратегию и достигать коммерческого успеха. Поэтому маркетинг является одной из основополагающих дисциплин в процессе подготовки профессиональных экономистов. Давайте же рассмотрим что из себя представляет маркетинг.

Маркетинг — это процесс планирования и воплощения замысла, ценообразование, продвижение и реализация идей, товаров и услуг посредством обмена, удовлетворяющего цели отдельных лиц и организаций.

Изучение рынка как такового, изучение потребителей, структуры фирм, товаров и анализ внутренней среды предприятий. Исследование, анализ и оценка нужд реальных и потенциальных потребителей продукции фирмы в областях, интересующих фирму. Маркетинговое обеспечение разработки новых товаров и услуг фирмы.

Выделяются четыре основных вида продвижения:

Реклама - любая информация, которая должна убедить потребителей сделать что-то, что нужно компании.

Связи с общественностью (PR) - это управление потоками между организацией и общественностью.

Стимулирование сбыта -  комплекс продвижения предмета маркетинга, представляющий собой систему побудительных мер и приемов, носящих кратковременный характер и направленных на поощрение покупки или продажи товара.

Личные продажи - инструмент, который используется для увеличения объема продаж непосредственно в ходе личного контакта. .

Независимо от того, продаете ли вы программное обеспечение, созданное вами самостоятельно, программный продукт для крупной корпорации или программное обеспечение как услугу (Saas), вам нужно следовать нескольким основным принципам, чтобы ваш продукт попал в руки клиентов. Создайте веб-присутствие и маркетинговый план, а также распространяйте информацию о своем программном пакете через блоги, сообщения на форумах и целевую онлайн-рекламу.

Подготовка программного обеспечения для торговой площадки

Многим понятно как работает продвижение таких товаров как например еда, для продукта делается упаковка которая в целом является самодостаточной системой продвижения, так как еда на полках продуктовых магазинах стоит рядом друг с другом и потребитель выбирает товар по его предпочтениям, понятно что большинство компании заказывают рекламу для своих продуктов и без этого не обойтись, если компания хочет сохранить свои продажи и выставить себя лучше по отношению к конкурентам.

Но как же тогда обойтись с программным обеспечением? Его не завернуть в обёртку и не пустить рекламу по телевизору, потому что большинство программного обеспечения приобретается онлайн, а не в живых магазинах. Из этого логично следует что для ПО и для других онлайн продуктов существуют уникальные правила продвижения и иногда они гораздо сложнее чем особенности продвижения прочих товаров.



1. Определите, что делает ваше программное обеспечение достойной покупкой
2. Определите аудиторию, которой вы будете продавать своё программное обеспечение
3. Размещение файлов программного обеспечения в облако, чтобы не занимать худшую пропускную способность.
4. Бета-тестирование вашего программного обеспечение, прежде чем показывать его клиентам

Маркетинг вашей программы



1. Создайте план, ориентированный на вашу целевую аудиторию.
2. Пригласите независимых рецензентов для использования и проверки вашего программного обеспечения.
3. Создайте присутствие своего программного обеспечения в социальных сетях и Интернете.
4. Устанавливайте конкурентоспособную цену на своё программное обеспечение, чтобы занимать продажи конкурентов.
5. Создайте бесплатную версию программного обеспечения, чтобы клиенты могли попробовать .
6. Напишите ключевые слова в копии вашего сайта, чтобы привлечь больше веб-трафика.
7. Чтобы привлечь потенциальных клиентов из онлайн-поисковых систем, попробуйте заполнить текст вашего сайта ключевыми словами, которые будут направлять людей на ваш сайт.
8. Ключевые слова должны быть специфичными для вашего программного обеспечения, но достаточно общими, чтобы люди, которые не ищут ваi конкретный продукт, вводили их в поисковые системы.
9. Предложите бесплатную пробную версию, чтобы пользователи могли ознакомиться с вашего программным обеспечением.
10. Скорректируйте свою маркетинговую стратегию после анализа показателей продаж программного обеспечения

**Продажа и поддержка вашего продукта**



1. Создайте интернет-магазин для продажи и поддержки вашего программного обеспечения.
2. Загрузите 1-2 минутное видео, описывающие функции вашего программного обеспечения.
3. Решите, делать ли вообще программное обеспечение с открытым исходным кодом или проприетарным.
4. Предлагайте надежную службу поддержки клиентов для поддержки вашего программного обеспечения.
5. Дружелюбное обслуживание клиентов также укрепит доверие ваших клиентов.
6. Если вы работаете в крупной компании, предлагайте круглосуточное обслуживание клиентов и поддержку программного обеспечения. Если вы занимаетесь небольшой операцией, пообещайте быстрое обслуживание клиентов в рабочее время.
7. Пообещайте возврат средств всем неудовлетворенным клиента.

## Шаблон приложения

Лучший способ представить своё приложение это предоставить покупателю его вид и описать его работу максимально понятным языком, но для интернет продуктов это сложнее так как к выбору программ для приобретения подходят очень аккуратно, так как за бесполезное или за плохо сделанное приложение могут потребовать больших денег.

Некоторые разработчики поэтому делают пробные версии своих приложений которые можно опробовать на определённый срок и впоследствии приобрести в личное пользование на неограниченное время.

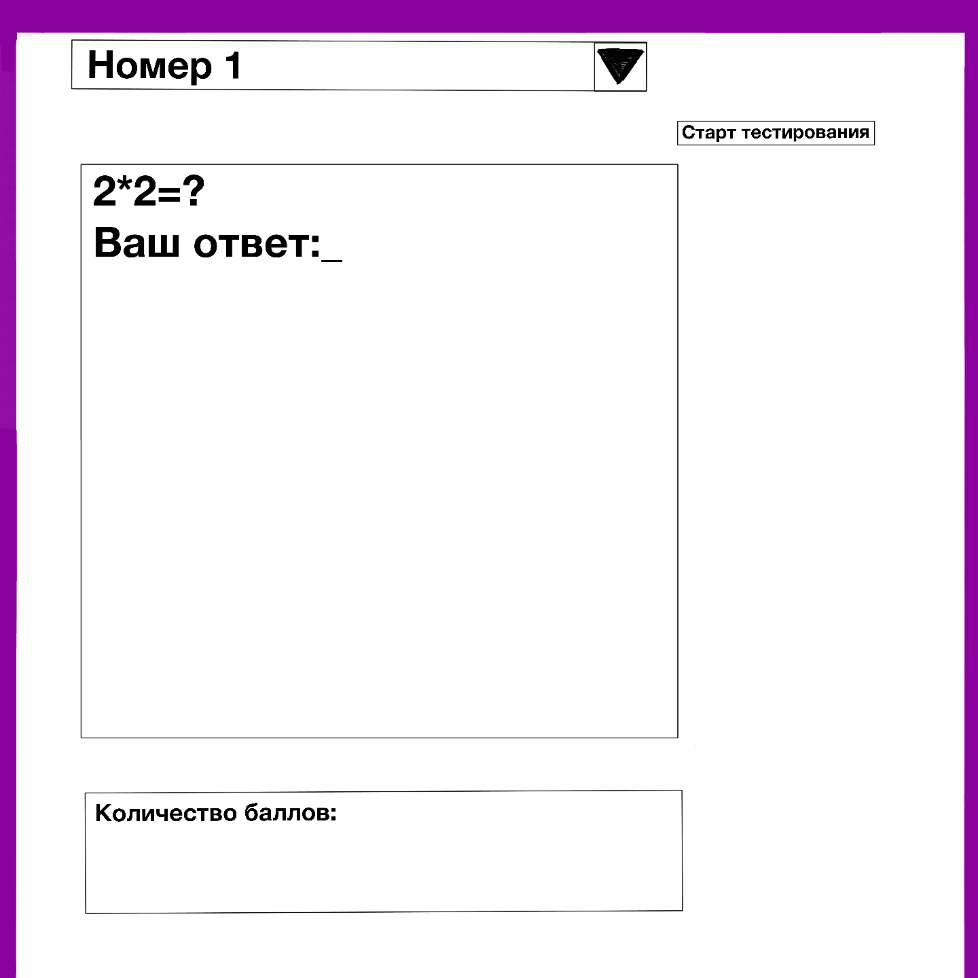


Рисунок 1. Шаблон приложения

**Создание сайта**

**Сайт**представляет собой совокупность логически связанных между собой страниц и функций. Его можно создать с помощью языка программирования html и используя стиль css.

Опросы и анкеты

Анкета (от франц. enquete - расследование) - опросный лист для получения каких либо сведений о том, кто его заполняет, или для получения ответов на вопросы, составленные по определенной программе.

**Принципы создание анкеты**

Первый принцип - Программная логика вопросов не должна быть смешиваема с логикой построения анкеты. Опросный лист строится под углом зрения психологии восприятия опрашиваемого.

Второй принцип - Непременный учет специфики культуры и практического опыта опрашиваемой аудитории.

Третий принцип - Одни и те же вопросы, расположенные в разной последовательности, дадут разную информацию.

Четвертый принцип - Смысловые "блоки" опросного листа должны быть примерно одного объема. Доминирование какого-то "блока" неизбежно сказывается на качестве ответов по другим смысловым "блокам".

Пятый принцип - Распределение вопросов по степени их трудности.

**Результаты опроса**

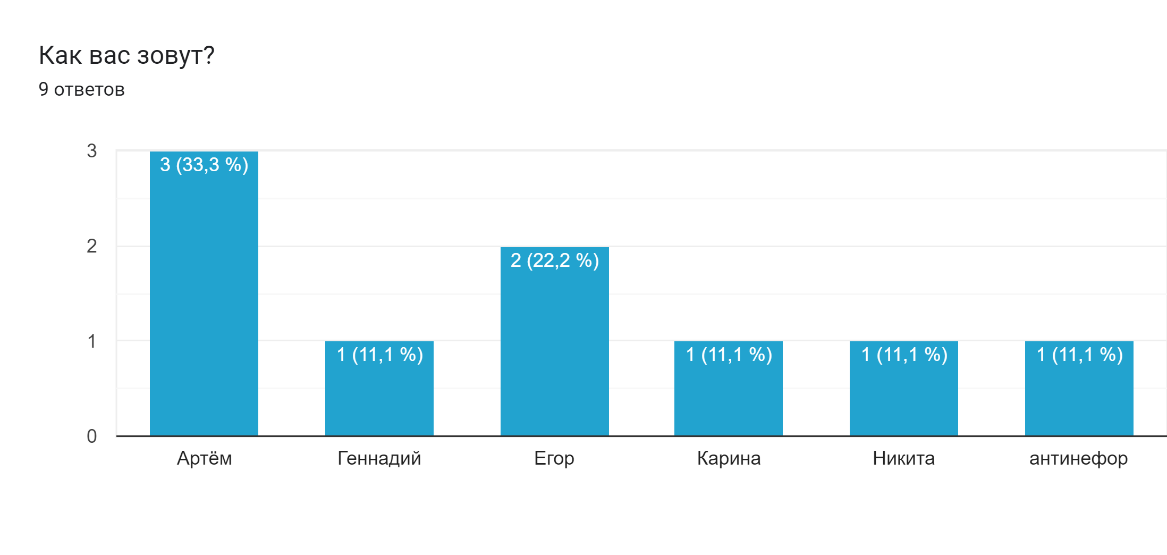


Рисунок 1. Опрос

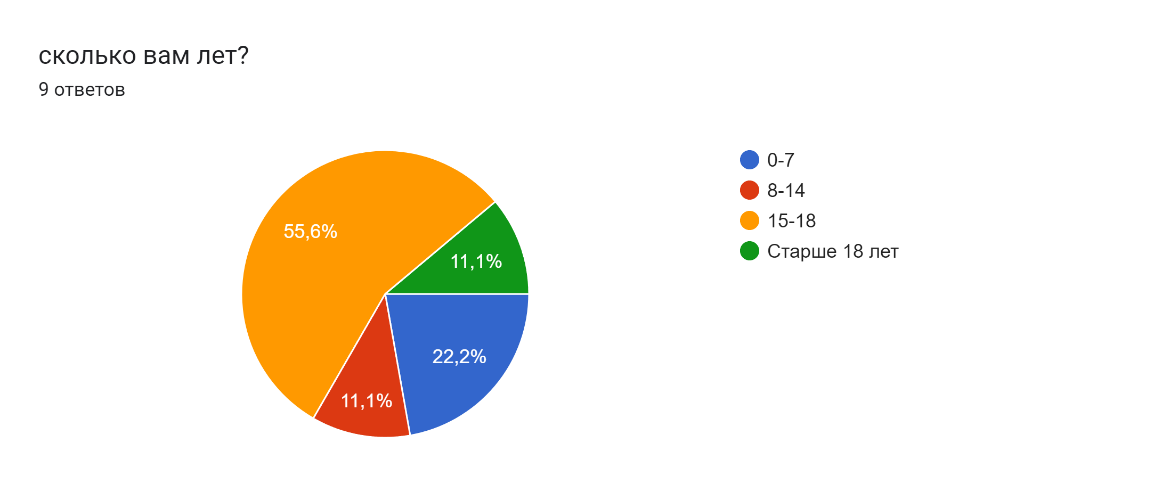


Рисунок 2. Опрос (возраст)

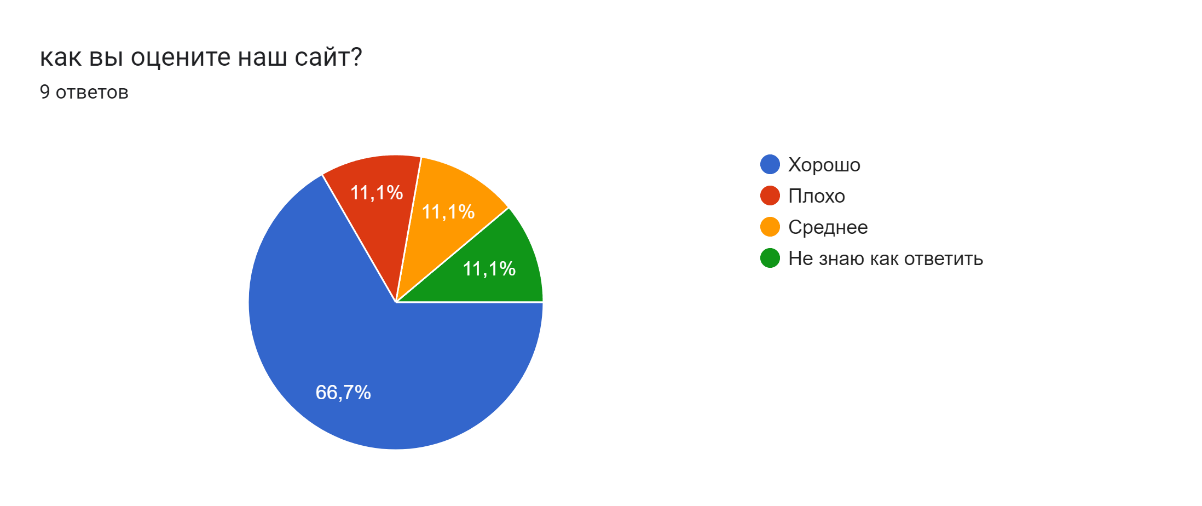


Рисунок 3. Опрос (оценка сайта)

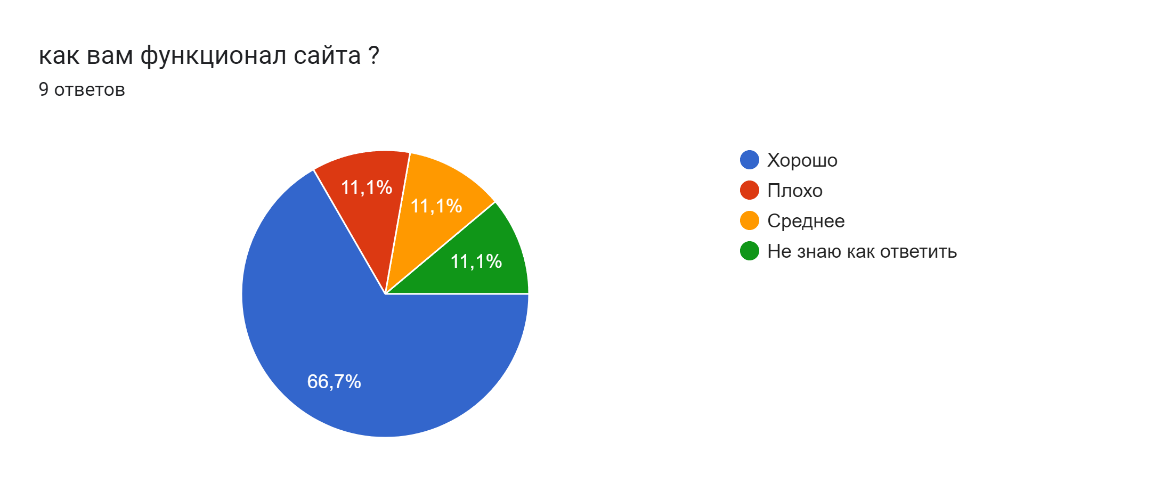


Рисунок 4. Опрос (функционал сайта)

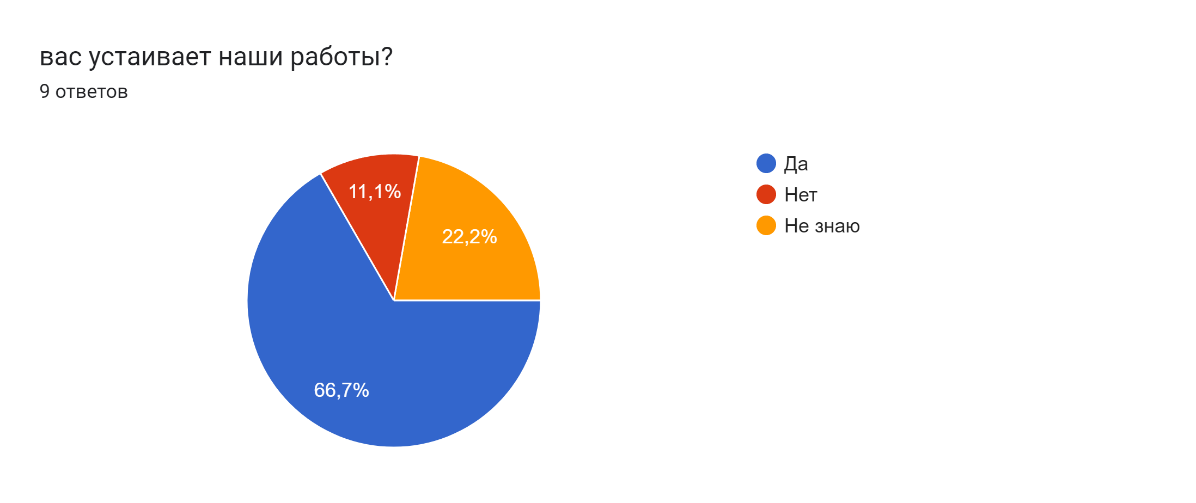


Рисунок 5. Опрос (оценка работы)

**Приложение**

1.<http://book.kbsu.ru/theory/chapter6/1_6.html>

2. [википедии](https://ru.wikipedia.org/wiki/Операционная_система)

3. [http://elibrary.sgu.ru/uch\_lit/1722.pdf](https://vk.com/away.php?utf=1&to=http%3A%2F%2Felibrary.sgu.ru%2Fuch_lit%2F1722.pdf)