**ВВЕДЕНИЕ**

**Актуальность работы.** Актуальность выбранной мною темы обусловлена постоянно растущим спросом на веб-сайты для компаний. На данный момент, практически все новые компании, стартапы имеют свой веб-сайт. Наличие такого в интернете даёт возможность получать клиентов удаленно.

**Цель работы.** Разработать программное обеспечение для компании.

Для достижения поставленной в работе цели были решены следующие задачи исследования:

* Изучение инструментов верстки, HTML, CSS.
* Изучение инструментов бэкенда, Python Django.
* Разработка веб-сайта.

**Объект исследования –** Веб-сайт на основе фреймворка Django

**Предмет исследования –** Разработка веб-сайта используя язык программирования Python

1 СПЕЦИАЛЬНЫЙ РАЗДЕЛ

1.1 Современные технологии, используемые для созданий веб-сайтов

Веб-сайт — это набор документов и файлов, которые лежат на каком-то сервере (компьютере) и объединены одним доменным именем. Доменное имя, это название веб-сайта, которое используется для его поиска в сети. Например «mail.ru» является доменным именем, которое объединяет странички, лежашие на сервере.

Сервером может является обычный компьютер, на котором установлены определенные программы, позволяющие работать веб-сайту. Браузер посылает определенный запрос, являющийся доменным именем веб-сайта, сервер его принимает и отправляет обратно некие документы. Это работает как сетевое копирование файлов. только по протоколу http.

Все документы веб-сайта, которые мы видим в браузере, написаны на языке НТМL. НТМL (HyperText Markup Language) это язык, являющийся стандартом для разработки документов, которые будут отображаться в ‘интернет-браузерах. Ключевой смысл языка НТМL. в разметке текста, для чего используются специальные команды, называемые теги, графически они отображаются в угловых скобках. НТМL документы хороши тем, что для их просмотра подходит практически любой браузер. Документ, написанный на языке НТМL, будет правильно отображаться на различных устройствах, имеющих различное разрешение экрана и операционную систему.  
 Практически все документы содержат в себе стандартные элементы, такие, как титульная надпись, заголовки или списки. С помошью тэгов НТМL можно обозначать данные элементы, обеспечивая веб-браузеры минимальной информацией для отображения данных элементов, сохраняя в пелом общую структуру и информационную полноту документов. Браузер это окно, через которое мы просматриваем интернет веб-сайты. То есть все странички должны быть написаны на языке НТМL. Создать документы, написанные на языке НТМL, которые станут основой вашего веб-сайта, можно несколькими способами, Основным способом является текстовый набор НТМL-кода в редакторе. Это достаточно трудоемкий процесс, который подразумевает глубокое знание принципов построения документов в НТМL, умение работать с каскадными таблипами стилей СSS, а также как минимум с javaScript. Вторым способом является использование СМS. Это движок веб-сайта, который позволяет пользователю почти автоматически создавать веб-сайты, подразумевает меньший набор знаний для создания и требуется гораздо меньшее зремя на разработку. Также при использовании СМS гораздо проше осушествлять дальнейшее обслуживание веб-сайта, добавление или удаление новой информации.

Также есть возможность создавать веб-сайты с помошью специальных редакторов, которые объединяются под общим названием WYSIWYG- «What You See Is What You Get». который можно перевести как «что ты видишь, то ты и получаешь» Создание веб-сайтов в этих программах напоминает работу в трафических и текстовых редакторах, но на выходе мы получим не ‘изображение, а документ на языке НТМL. Этот метод позволяет достаточно быстро получить документ с достаточно сложным оформлением.

Рассмотрим достоинства и недостатки этих методов. По началу кажется, что достоинства СМS или редакторов WYSIWYG неоспоримы, но это далеко не так. Веб-сайты, написанные вручную, хоть и требуют времени на создание, но зато получаются качественней.

Обозначим преимущества веб-сайтов, написанных вручную:

Созданный таким образом веб-сайт гораздо сложнее взломать, так как нет Mysql базы данных и отсутствует файл конфигурации, а именно с помошью эти файлов в первую очередь хакеры пытаются взломать веб-сайт.

Веб-сайты, написанные на НТМL, если сравнивать их с СМS системами отображаются гораздо быстрее. Так получается потому. что сами файлы получаются намного компактнее и не надо делать запросы в Mysql базу данных.

Скорость отображения веб-сайта положительно влияет на позицию выдачи при поисковых запросах. При одинаковых условиях веб-сайт. написанный на НТМL. будет выше.

Веб-сайт занимает меньше места, что позволяет экономить на тарифах хостинга. НТМL веб-сайт не будет содержать продублированных страничек, что положительно отображается на результатах понска. Для описания внешнего вида элементов применяется каскадные таблипы стилей СSS.

СSS довольно элементарный формальный язык, который придумали для описания внешнего вида документов. Это говорит о том, что он довольно прост и состоит из самобытных примитивных конструкций, которые не так сложны для изучения. Самое сложное не синтаксис, не правила написания конструкций, а огромное количество СSS свойств для запоминания, которые выполняют различные задачи. Благо все правила англоязычные с соответствующей смысловой нагрузкой. Простой перевод на наш язык дает понятие о том, что это правило делает и наоборот - при переводе того, что мы хотим добиться определенным свойством на английский язык, велика вероятность того, что мы получим правильное свойство. Это значительно упрошает запоминание СSS правил на интуитивном уровне. Например, если нужно задать фоновый цвет достаточно сделать перевод на английский, в результате чего получаем background-соlоr (отдельные слова в СSS пишутся через дефис.

СSS довольно просто использовать в НТМL документах. Его можно подключить как внешний СSS файл к документу.

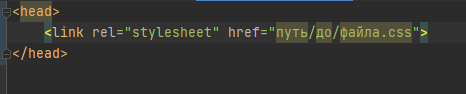


Рисунок 1.1 – Подключение CSS файла в HTML странице

Это наиболее распространенный способ подключения таблиц стилей к документу. когда внешнее оформление страниц выносится в отдельный внешний СSS файл.

СSS имеет довольно простой синтаксис. Правила объявлений настолько просты, что их можно описать в одном предложении. Сначала пишется селектор. который выбирает конкретный элемент на странице, после фигурных скобок пишутся свойства со значениями после двоеточия, а сами свойства отделяются друг от друга точкой с запятой. Самое сложное в СSS объявлении - селектор.

СSS селектор - (от слова select - выбирать) - это конструкция, с которой начинается каждый блок объявлений, н которая служит для выборки элемента или однотипных элементов на странице для дальнейшей стилизации. Чаще всего в качестве селектора используется определенный класс тега.



Рисунок 1.2 – Подключение CSS файла в HTML странице

Здесь селектором выступает класс my-class тега div. который получает необходимое оформление в СSS файле. В данном случае фоновый пвет - серый. Соответственно, если на странице есть несколько тегов (не только 4%) с классом my-class. все эти элементы получат одинаковое оформление серый фон пвета #999.

1.2 Этапы создания веб-сайта

Первый этап, создается модель или прототип веб-сайта.

На этом этапе примерно набрасывается внешний вид веб-сайта и обшая структура веб-сайта, определяется какие элементы и в каком порядке он будет содержать. Для этого могут быть использованы различные программы и онлайн сервисы, но также можно просто отобразить на листе бумаги.

Второй этап, создание дизайна веб-сайта.

Как правило, макет, отражающий дизайн веб-сайта, рисуется в трафических редакторах, таких как Adobe Photoshop или Gimp или Adobe Experience. Это является одним из главных этапов создания веб-сайта, именно на этом этапе окончательно формируется внешний вид веб-сайта. Добавляется вся графика, уточняются и прорисовываются все детали

Третий этап, верстка страниц в НТМL.

Макет веб-сайта, нарисованный в графическом редакторе, разбивается на отдельные графические элементы и описываются на языке НТМL, также при верстке могут быть использованы такие технологии, как СSS и javascript, если необходима некая динамичная подгрузка или анимация.

Четвертый этап, разрабатывается программная часть или используется движок.

Пятый этап, веб-сайт размещается на хостинге.

Подготовленные НТМL странички копируются на сервер. Им присваивается определенное доменное имя и получают публичный доступ.

1.3 Дизайн веб-сайта

Первое, на что обрашает внимание человек, когда заходит на какой- либо веб-сайт, это его дизайн. Именно от того, насколько удачно сделан дизайн веб-сайта зависит захочет ли пользователь просматривать страницу дальше или закроет ее. Поэтому очень важно при проектировании веб-сайта хорошо понимать на какую аудиторию он рассчитан, какие потребности возникают у пользователей. Всеми этими вопросами занимается UX/UI. UX или User Experience (дословно: «опыт пользователя») — дисциплина. изучающая опыт взаимодействия пользователя с продуктом, восприятие и UI.  
 UI - это user Interface (дословно «пользовательский интерфейс») — то, как выглядит интерфейс и то. какие физические характеристики приобретает. Определяет, какого вета будет ваше «изделие», удобно ли будет человеку попадать пальцем в кнопочки, читабельным ли будет текст и тому подобное. UX - это понятие, которое применимо не только в веб-дизайне, но и других областях. Опыт взаимодействия - это когда специалист на заводе управляет сложной техникой посредством нажатия пары кнопок, при чем кнопка отключения значительно крупнее. UX - это когда вы срезаете путь до дома тропинкой, хотя есть красивый и чистый тротуар: это когда вы ставите стаканчик с кофе в специальные углубления на панели в салоне вашего авто: это когда педаль тормоза шире педали газа. Все это пользовательский опыт взаимодействия. Примеров может быть масса, но суть одна: UX - это способ достижения пели пользователем максимально комфортным способом. И при создании хорошего дизайна необходимо учитывать это. UX - это область, в которую входит как исследование и проектирование, так и визуализация, и верстка. В результате работы над UX мы должны получить рабочий прототип. Так как речь идет о веб-дизайне, таким прототипом является HTML вёрстка. UX в веб-дизайне - это фундамент. на котором строится дизайн веб- сайта или приложения Это проектирование интерфейса с учетом потребности людей, для которых вы создаете дизайн веб-сайта. Кто конечный пользователь, какие пели человек преследует, используя ваш интерфейс, как помочь ему максимально быстро и комфортно получить результат. Вот основные задачи, которые решает UX.

Многие веб-дизайнеры путают веб-дизайн и визуальный дизайн. Кто-то до сих пор считает, что процесс создания дизайна веб-сайта - это, в первую очередь, создание картинок, кнопочек, иконок (UI), совершенно забывая о фундаменте (UX). о исследовании людей, для которых все это рисуется.   
 Задачи, решаемые UX.

Главная задача дизайнера - максимально повысить уровень удовлетворенности конечного пользователя от взаимодействия с продуктом. Под продуктом мы будем понимать любой объект взаимодействия с пользователем, будь то реальный продукт, услуга, веб-приложение или веб-сайт. Задача веб-дизайнера - проложить мост между владельцем бизнеса и потенциальным клиентом. Необходимо понимать, что основная работа по исследованию при проектировании UX - это изучение потенциальной аудитории продукта.

Самое сложно здесь - это то, что нельзя пощупать, нельзя измерить - это эмпатия веб-дизайнера или UX специалиста, способность понять желания н чувства других людей. Right Way - исследование. Исследование - это самый первый шаг дизайнера при работе над любым проектом. Нужно чётко вывести группу пользователей, максимально узнать аудиторию. под которую вы разрабатываете интерфейс, узнать потребности этой аудитории. Основываясь только на своем личном опыте и своих ощущениях довольно сложно прийти к объективно правильному решению в интерфейсе, ведь мы не беспристрастны по отношению к результату вашего творчества и ваш личный опыт относительно конкретного продукта может быть весьма ограниченным. Тем не менее, если вы хорошо изучили продукт, для которого делаете дизайн веб-сайта, то можете сделать какие-то выводы, принять решения. Самый минимум, который вы просто обязаны сделать это изучить текущий опыт бизнеса, накидать портрет среднего потребителя продукта. Этот пункт говорит о том, что проектирование UX очень тесно связано с бизнесом. В начале своего исследования нужно пообщаться с владельцем бизнеса, нашим заказчиком. Он является специалистом в своем продукте и у него есть опыт работы с клиентами, он знает свою аудиторию. Необходимо записать все ключевые моменты для дальнейшего анализа. Также нужно изучить опыт аналогичных продуктов. После того. как вы определили аудиторию. определили пели и ценности персон, можно приступить к моделированию каркаса интерфейса, так называемому Вайрфреймингу. Он служит для распределения информации на будущих страницах по степени важности - сверху вниз. В процессе моделирования необходимо продумать какая информация будет на странице, определить базовую форму вывода информации, но не вдаваться в детали, не заниматься визуализацией. Для создания каркаса можно воспользоваться специальным приложением или нарисовать от руки на бумаге в клеточку или в точечку, после чего отсканировать н положить в папку с материалами проекта. Отлично подходит онлайн приложение wireframing.cc, так как можно в реальном времени вносить правки и согласовывать с клиентом.

Нужно проработать все страницы веб-сайта или приложения и только после того, как каркас всего веб-сайта готов, переходить к следующему шагу UI. Визуализация после того, основательно проработан UX, создано несколько каркасов и выбрали наиболее привлекательные варианты, можно приступать к визуализации. Визуализация - это прорисовка каркасов.Ссоздание единого стиля, оформление контента. Другими словами, мы начинаем работать над UI. Чаше всего для визуализации используется Adobe Photoshop, Scetchapp, Inkscape+Gimp или другие инструменты. Для работы с векторной графикой будем использовать Inkscape — для создания иконок и другой необходимой графикой, для работы с растровой графикой – лучшее решение, это Adobe Photoshop.

В веб-дизайне есть правила хорошего тона. Разрабатывая дизайн веб-сайта веб-дизайнер придерживается определенных негласных правил при оформлении. Межстрочный интервал, отступы, шрифты, размеры элементов - это все поддается измерению и есть откровенно плохие реализации. Поэтому можно составить свод правил. придерживаясь которых поможет не сделать плохую визуализацию вашего дизайна. Перечислим некоторые правила:

Типографика, текст, ссылки:

- Не использовать слишком большие заголовки.

- Не использовать шрифт меньше 12px.

Не делать слишком маленький или слишком большой межстрочный интервал - не растягивать буквы инструментом "Transform", шрифт должен быть естественно пропорциональным:

- Не использовать больше 3 шрифтов на странице.

- Не использовать слишком маленький контраст, не печатать светло-серым по белому или тёмно-серым по чёрному.

Использовать интервал между символами с осторожностью если знаете, что делаете и выбранный шрифт позволяет сделать текст "воздушным" наиболее элегантно:

- Не делать слишком маленьких отступов между абзапами, заголовками и элементами, что называется «дать воздуха дизайну»:

- Не использовать верхний регистр букв без необходимости. Не использовать для основных текстовых блоков слишком сложный декоративный шрифт, это должен быть простой и легко читаемый шрифт какого-либо семейства Sans или Serif.

Все ссылки, за исключением пунктов навигации, должны быть подчеркнуть. Старайтесь также оформлять ссылки, которые уже были посещены, более темным цветом, в отличие от цвета ссылок по умолчанию: Если иерархия веб-сайта содержит более 3-х уровней, не забывайте о так называемых хлебных крошках. Это пункты в меню, которые подскажут пользователю на каком уровне он находится графика, иконки, фотографии:

- Не использовать в дизайне шаблонные фотографии Лучше сделать самостоятельно, порекомендовать заказчику обратиться к фотографу или найти наиболее "жизненные" фотографии.

- Не использовать иконки, сделанные из фотографий.

- Все иконки должны быть выполнены в едином стиле.

- Не увеличивать фотографию больше ее оригинального размера.

- Не масштабировать графику непропориионально.

Не применять режимы наложения слоев. отличные от обычного (Normal). Стараться не применять фильтры на изображения, которые должны иметь несколько состояний (обычно и при наведении, например). Все наложения. изменения - только посредством наложения нового слоя. Всё должно быть легко воспроизведено в HTML верстке. К ретуши и подготовке фото это не относится.

Не масштабировать фотографию до конвертации в смарт объект.

Обрезать фотографию только посредством обтравочной маски для фигуры. оригинал фотографии. как и в предыдущем пункте должен быть сохранен в смарт объект.

Если иконки рисуются в Photoshop. не растрировать их в уменьшенном размере. Не нужно забывать, что макету еще предстоит эта верстки и все иконки необходимо будет векторизовать. Оригинальные смарт объекты с иконками должны быть достаточно крупными для качественной трассировки.

В идеале, у вас должны быть все плоские иконки и графика в формате SVG и в отдельной папке Цвет.  
 - Не использовать чистые вета, обязательно стараться добиться наиболее сприятного оттенка.

- Не использовать более 2-х акцентных иветов на странице и не более двух темно-серых (или черного) иветов для текста. В идеале - только 1 акцентный цвет 1 темно-серый/черный для текста.

Стараться закрашивать акцентным иветом только те элементы, которые наиболее важны на странице, акцентировать на них внимание. Это кнопки, стрелочки. галочки важных пунктов, текстовые ссылки. Информативные иконки (мелкие иконки типа "логин". "пароль "и иконки в формах акцентировать не обязательно).

Правила работы в графическом редакторе, организация работы и прочее:

- Называть слои со смыслом.

- Стараться упорядочивать смысловые блоки и составные элементы интерфейса в группы. Всегда создавать отдельную папку со всеми используемыми шрифтами в формате ТТF или ОТF. Иконочные шрифты также должны быть размешены в этой папке. Создавать макеты 1 в 1 (72 пикс. на дюйм. При 100% отображении макеты должны быть в таком же масштабе, в котором предполагается результат НТМL верстки. Обязательно использовать в работе какую-либо систему сеток Можно разработать самому. но рекомендуется использовать систему сеток Bootstrap. Ширину контента можно настраивать вручную от оригинальной ширины Bootstrap сетки, для этого можно использовать плагины для создания сеток в случае, если пользоваться Photoshop. Не создавать декоративные элементы, если в этом нет практического смысла. Если это имиджевый веб-сайт и нужна красивая картинка, данное правило можно опустить и придерживаться смысловой визуальной иерархии. Использовать правило "внутреннего и внешнего”, которое гласит, что расстояния между внутренними элементами блоков должны быть меньше, чем внешнее расстояние между блоками.

Не забывать, что люди чаше всего приходят не на главную страницу веб-сайта. а на внутренние. поэтому нужно продумать информативные универсальные блоки - шапку. подвал, сайдбары (если есть). Шапка должна быть максимально информативной, но не перенасыщенной. Обязательные элементы: Логотип, название проекта, навигация. Поиск по веб-сайту и другие элементы размешаются в зависимости от проекта. Хорошее решение - размещение в футере развернутой навигации или карты веб-сайта со всеми потаенными местами. Нужно постараться продумать подвал веб-сайта до мелочей, стараться не делать скудных узких подвалов с логотипом и номером телефона. Лучше делать его высоким и развернутым. Очень часто пользователи находят нужную потаенную информацию именно в подвале веб-сайта.

1.4 Программное обеспечение, необходимое для создания веб-сайта

1.4.1 Утилиты

Для того чтобы автоматизировать процесс веб разработки понадобиться такая утилита как gulp. Gulp - это инструмент, который помогает автоматизировать рутинные задачи веб-разработки. Gulp предназначен для решения таких задач, как:

- Создание веб-сервера и автоматическая перезагрузка страницы в браузере при сохранении кода, слежение за изменениями в файлах проекта. Использование различных Javascript, CSS, HTML препроцессоров, (CoffeeScript, Less, Sass, Stylys, Jade итд)

- Минификация СSS и JS кода, а также оптимизация отдельных файлов проекта в один.

- Автоматическое создание вендорных префиксов (приставок к названию CSS свойства, которые добавляют производители браузеров для нестандартных свойств) для СSS.

- Управление файлами и папками в рамках проекта - создание, удаление и переименование.

Запуск и контроль выполнения внешних команд операпионной системы. Работа с изображениями - сжатие, создание спрайтов, ресайз (png, jpg, svg идр.) Деплой (отправка на внешний сервер) проекта по FТР, SFТР, Git.

Подключение и использование в проекте безгранично большого количества Node.js. и Gulp утилит. программ и плагиновСоздание различных карт проекта и автоматизация другого ручного труда.

Можно с уверенностью сказать, что Gulp и множество утилит, написанных для него, подходят для решения практически любой задачи при разработке проекта любой сложности - от небольшого веб-сайта до крупного проекта. Любой проект, использующий Gulp имеет в корне файл gulpfile.js, который содержит набор инструкций по управлению проектом. Сразу хочется сказать что написание инструкций для gulp не является программированием, хотя пишутся на языке Javascript. Для установки gulp необходимо установить node.js. Node.js - программная платформа, основанная на движке V8 (транслирующем Javascript в машинный код). превращаюшая javascript из узкоспециализированного языка в язык общего назначения. Node.js добавляет возможность Javascript взаимодействовать с устройствами ввода вывода через свой АРI (написанный на С++). подключать другие внешние библиотеки, написанные на разных языках, обеспечивая вызовы к ним из Javascript. Может применяется преимущественно на сервере, выполняя роль веб-сервера. При создании дизайна веб-сайтов в Adobe Photoshop не обойтись без использования какой-либо системы сеток. Можно создать сетку самостоятельно, используя направляющие, либо воспользоваться уже готовым, проверенным решением - готовой сеткой Bootstrap в формате РHР. Несмотря на то, что размеры Bootstrap сетки несколько стандартизируют дизайн, данный вариант успешно используют в своей работе огромное количество веб-дизайнеров по всему миру. Кроме того, такая стандартизация позволяет максимально быстро выполнять НТМL. верстку созданного макета, что неминуемо приводит к ускорению работы над проектом в условиях разработки веб-сайтов на потоке или создания шаблонов. Браузеры, такие как Opera, Google chrome, Mozilla Firefox для тестирования правильного отображения веб-сайта в различных средах.

1.4.2 Текстовые редакторы

Эти программы будут нужны для написания и редактирования текстовых документов, коими и являются HTML документы. Здесь подойдет даже самый простой редактор «Блокнот», но для удобства лучше воспользоваться специальными программами, которые предназначены для написания HTML кода. Такие программы подсвечивают теги, помогают находить ошибки в коде и лучше структурировать код. Также в них есть множество полезных функций, позволяющих ускорить процесс написания кода. Примерами таких программ являются Notepad++, Aptana Studio, Sublime text. Для наших целей отлично подойдет Sublime text, так как у него приятный интерфейс. очень гибкая настройка и присутствует функция множественного выделения, которая очень ускоряет процесс верстки.

1.4.3 Графические редакторы

При создании прототипа и макета веб-сайта можно воспользоваться графическими редакторами, это программы, позволяющие работать с графикой (создавать и изменять различные изображения). Также, как и текстовые редакторы, они бывают простыми и сложными. К простым в первую очередь относят Windows Paint. К сложным графическим редакторам, которые работают с растровой графикой относится Adobe Photoshop и Gimp. Для векторной графики используется Inscape, Adobe Illustrator или CorelDraw. Также можно использовать онлайн-сервис wireframe.cc для создания прототипа веб-сайта. В нем можно набросать структуру веб-сайта, создания прототипа веб-сайта, обозначить расположение ключевых элементов. их размеры. Примерные цвета.

1.5 Python – язык программирования

Python - это высокоуровневый язык программирования, который используется в различных сферах IT, таких как машинное обучение, разработка приложений, web, парсинг и другие. Python смог захватить малую часть рынка веб-разработки, иногда используется для написания десктопных приложений и, конечно, тотально доминирует в сфере машинного обучения. Кроме того, на нём создаётся много прототипов, которые позволяют быстро набросать функционал и внешний вид будущего проекта. Back-end сайта для разработки серверной части сайта используются фреймворки: Django и Flask. Они превращают Python в серверный язык программирования, возможности которого не уступают другим популярным инструментам. Программист легко может работать со связями URL адресов, обращениями к базам данных и созданием HTML файлов, которые пользователь видит в браузере. И хотя большую часть рынка серверной веб-разработки контролирует PHP, всё больше программистов отдают своё предпочтение разработке на Python.

1.5.1 История развития Python

Язык начал разрабатывать программист,Гвидо ван Россумом, в конце 1980-х. На тот момент он работал в центре математики и информатике в Нидерландах. Гвидо ван Россум увлекался работой с «железками» ещё со школьных лет, и хотя он не находил поддержки и одобрения у своих сверстников, это не помешало ему самостоятельно разработать язык программирования. Россум работал над Python в свободное время, в качестве основы он взял язык программирования ABC, в разработке которого когда-то участвовал. Этапы истории языка программирования Python:

- В феврале 1991 исходный код языка был опубликован на alt.sources. Уже тогда язык придерживался объектно-ориентированного подхода, мог работать с классами, наследованием, функциями, обработкой исключений и всеми основными структурами данных.

- В 2000 году вышла в релиз вторая версия Python. В неё добавили много важных инструментов, включая поддержку Юникода и сборщик мусора.

- 3 декабря 2008 в релиз вышла третья версия Python, которая является основной до сих пор. Многие особенности языка были переделаны и стали несовместимы с предыдущими версиями. И хотя функциональность третьей версии ничем не уступает второй, развитие языка разделилось на две ветки. Кто-то продолжал использовать Python 2, чтобы поддерживать старые проекты, кто-то полностью перешёл на третью версию.

- Дату смерти второй версии установили на 2015 год, однако, боясь не успеть перенести весь существующий код на Python 3, время жизни Python 2 продлили жизнь до 2020 года.

1.6 Клиент-серверная архитектура

Клиент – компьютерное устройство, которое отсылает запросы серверу, касающиеся выполнения определенных задач или предоставления конкретной информации. Сервер – компьютерное устройство, гораздо мощнее обычного ПК.

Система работает по следующему принципу:

- Клиент отправляет запрос серверной машине.

- Сервер принимает обращение с требованием выполнить определенное действие и выполняет поставленную задачу.

- Программно-аппаратный комплекс отправляет клиенту результат выполненной работы, обработанного запроса.

Модель клиент-сервер предоставляет возможность разграничить поставленные задачи и работу над вычислениями между теми, кто заказывает услуги и теми, кто их поставляет. Основные компоненты системы:

- Клиент. Рабочая станция считается входной точкой конечного пользователя в данной системе. Отправляет запросы, получает ответы.

- Сервер. Взаимодействует с многочисленными клиентами и решает поставленные ими задачи.

- Сеть. Здесь происходит передача данных. Посредством сети можно соединить рабочие машины общими ресурсами.

- Приложения. Могут обрабатывать информацию, организовывать физическое распределение данных между сервером и клиентом. Программным обеспечением оснащают серверные устройства для сбора данных, работы с ними и хранения. А также ПО устанавливают на компьютерной станции-клиенте.

О технологии клиент-сервер.

Серверное устройство поддерживает многопользовательский режим и обеспечивает одновременно работу с несколькими клиентами. Конечно, машина не может решать в прямом смысле слова одновременно несколько поставленных задач, она выстраивает запросы в очередь по мере поступления, обрабатывает обращения и отправляет результаты работы. Запросы можно выстраивать в списке по приоритетности. Чем важнее запрос, тем быстрей его обрабатывают, даже, если он поступил позже.

Рядовые пользователи сети интернет даже не догадываются о том, как их запросы моментально обслуживаются, чтобы они читали новости, книги, тематические статьи, смотрели интересные видео и фильмы, ходили по форумам, «зависали» в социальных сетях, оплачивали счета, общались с друзьями, оформляли заказы на покупку товаров и т.д. Главное, что ответная реакция быстрая.

Именно технология клиент сервер предоставляет возможность реализовать вышеуказанные многочисленные поставленные задачи. Обычно клиент – это браузер конкретного пользователя. А серверами зачастую выступают:

- Любые серверы http.

- Наборы серверных машин (например, Denwer).

- Локальный веб-сервер.

Обмен информацией между клиентом и сервером происходит благодаря сетевым протоколам в интернете. Каждой услуге соответствует определенный протокол, их предостаточно. Запросы, отсылаемые клиентом, классифицируют как http сообщения. Здесь четко указано, какие сведения нужно предоставить, в каком оформлении. Серверное устройство после анализа и обработки запроса, обычно отвечает html документом – дает свой http ответ.

Сообщение от клиента поступает с дополнительными данными, чтобы серверу было понятно, как с ним работать. Ответ машины также отправляется с кодами помимо полезных запрашиваемых данных, чтобы браузер оценил понятливость аппаратно-программного комплекса при обработке его запроса. Смотря на каком уровне осуществляется взаимосвязь клиента с сервером, отсылаемые сообщения браузером упаковываются по-разному. Как будто они оборачиваются клиентом в несколько слоем обертки. После того, как послание поступило серверной станции, она приступает к разворачиванию всех этих слоев, проводит анализ информации и сбор данных.

Говоря больше о технологии клиент-сервер, следует уточнить, что браузер первый выходит на контакт и делает запрос серверной машине, которая лишь предоставляет услуги в ответ на сообщения и указывает, какие условия нужно при этом соблюдать. Разные компьютерные устройства используют, чтобы установить программное обеспечение клиента и серверного оборудования. Но есть случаи, когда они работали на одном ПК.

Когда на одном сайте одновременно находятся несколько посетителей, к серверу в один момент обращается много клиентов. Однако одномоментное поступление запросов ограничено мощностью и возможностями серверных устройств, а также характером отправляемых сообщений.

Архитектура клиент-сервер.

Благодаря архитектуре клиент и сервер определены позиции взаимной связи между компьютерными машинами лишь в целом. Что же касается нюансов взаимодействия, они определены протоколами. Технология вполне прозрачно намекает на разделение в сети рабочих машин: серверы и клиенты. Рабочий контакт всегда инициирован клиентской машиной. Протокол же описывает, по каким правилам этот контакт установлен и действует. Архитектура взаимодействия между клиентом и сервером подразделяется на два вида:

- Двухзвенная. Сторонние ресурсы не задействованы. Одна машина обрабатывает поступившие сообщения. В этом случае сервер должен быть высокопроизводительным. Несмотря на эти жесткие требования, архитектура очень надежная. Первый уровень - клиент отправляет запрос. Второй уровень - сервером принимается сообщение, обрабатывается и отправляется ответ.

- многоуровневая. Речь идет о любой современной архитектуре СУБД. Принципиальное отличие и особенность: запросом клиента занимаются одновременно несколько серверных устройств. Операции перераспределяются, нагрузка на серверную машину снижена и оптимальная. Единственный минус: низкая надежность по сравнению с предыдущим вариантом.

1.7 Постановка задачи дипломного проекта

Дипломный проект на тему «Разработка web-сайта для ТОО “Ecology Expert”» содержит в себе следующие задачи:

* изучить теоретические и практические материалы о создании сайтов;
* выбор программного обеспечения для реализации практической части дипломного проекта;
* проектирование страниц сайта;
* анализ и подсчет необходимых таблиц базы данных;
* 123123сбор информации из открытых источников для защиты локально вычислительной сети;
* 123123обеспечение безопасности смоделированной локальной сети;
* 123123повторная атака на защищенную сеть;
* 123123сравнение сохранности локально вычислительной сети до и после защиты1.

1.7.1 Цель дипломного проекта

Целью дипломного проекта является систематизация и углубление знаний  
по моей специальности, а также разработка web-сайта для компании “Ecology Expert” посредством использования фреймворка Django.

1.7.2 Актуальность дипломного проекта

Актуальность выбранной мною темы обусловлена постоянно растущим спросом на веб-сайты для компаний. На данный момент, практически все новые компании, стартапы имеют свой веб-сайт. Наличие такого в интернете даёт возможность получать клиентов удаленно. Предоставляя информацию о себе, компания также может получить обратную связь по своим продуктам, а также в интернете легче организовать таргетированную рекламу. Таргетированная реклама – реклама, направленная на определенный сегмент пользователей, фильтрованный по возрасту, полу и другим признакам.

1.8 Программное обеспечение Pycharm

Чтобы разрабатывать веб, десктоп и консольные приложения нужно программное обеспечение, отвечающее некоторым запросам. Среди них я выделяю:

- Интуитивно понятный интерфейс. Удобный и понятный интерфейс обеспечивает большую скорость разработки за счёт повышения скорости навигации по интегрированной среде разработки.

- Подсветка различных структур данных разными цветами. Для примера все внутри строки помечается зеленым цветом, ключевые слова оранжевым (см. рис. 1.3). Человек получает большую часть информации глазами, а значит, что такая подсветка сделает процесс разработки быстрее и приятнее.

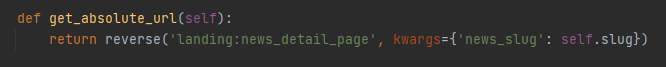


Рисунок 1.3 – Демонстрация подсветки IDE

- Возможность быстрого переключения между вкладками. В случае, если ведется работа над двумя или более файлами одновременно, гораздо удобнее работать над ними если можно быстро переключаться с одного на другой. Это распространенная функция среди всех современных интегрированных сред разработки (см. рис. 1.4).



Рисунок 1.4 – Пример вкладок в приложении Pycharm

- Подсказки. Подсказки становятся необходимыми в больших проектах, ведь запомнить каждую переменную или класс становится просто невозможно, поэтому эта функция также присутствует в большинстве современных интегрированных сред разработки. При этом, в Pycharm помимо подсказываемого названия объекта, указывается его тип (см. рис. 1.5).

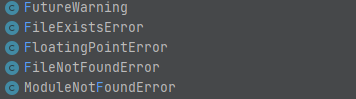


Рисунок 1.5 – Демонстрация подсказок в Pycharm

Так как Pycharm отвечает моим требованиям я решил выполнять поставленные задачи на нем.

1.8.1 Установка Pycharm

Pycharm это интегрированная среда разработки созданная разработчиками JetBrains под язык Python. Так как разработчики являются частью коммерческой компании, есть два вида, в котором поставляется приложение.

Первый, это платная версия программы с 30-дневным бесплатным пробным сроком, который после нужно оплатить на год.

Второй, это бесплатная версия, но с урезанным функционалом. В бесплатной версии отсутствует поддержка веб-разработки, баз данных и SQL в целом.

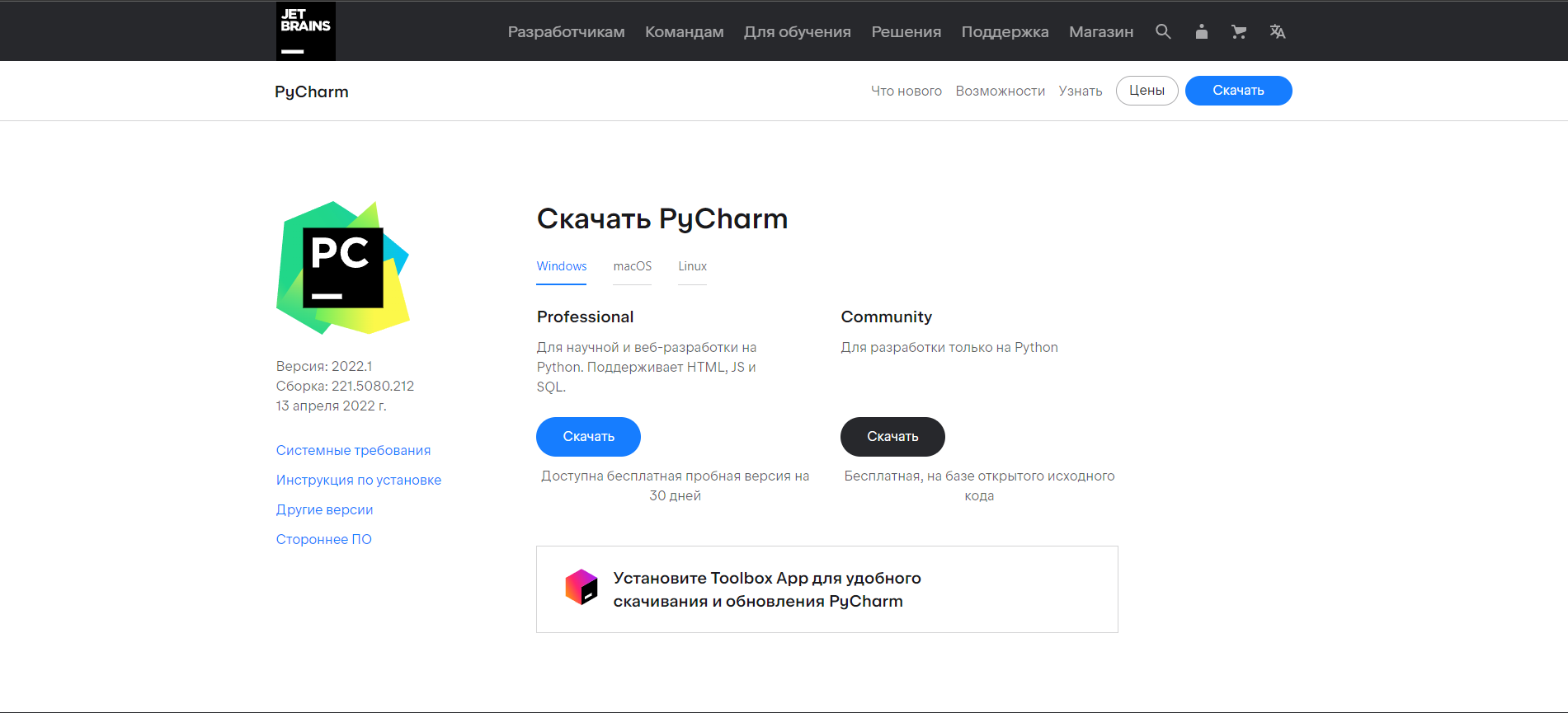


Рисунок 1.6 – Страница загрузки приложения

Я выбрал для себя профессиональную версию приложения, так там есть нужный мне функционал, после скачивания установщика начинается установочный процесс (см. рис. 1.7).

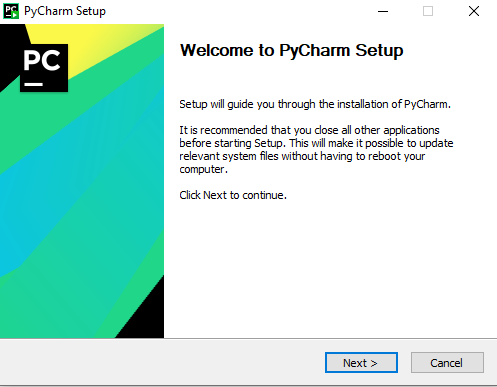


Рисунок 1.7 – Установка приложения

По завершению установки откроется приветственное окно (см. рис. 1.8)

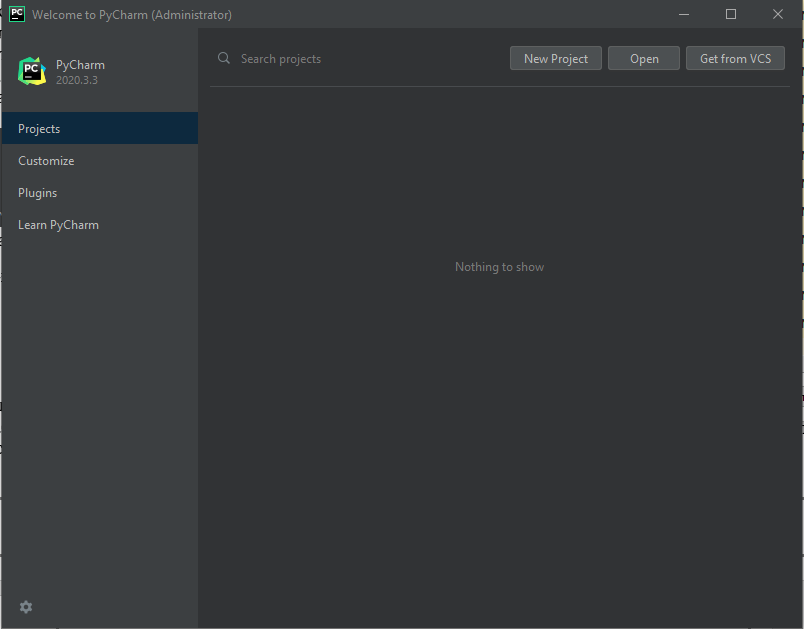


Рисунок 1.8 – Приветственное окно

1.8.2 Создание проекта

Проект создается по нажатию на соответствующую кнопку. После которой открывается выбор для создания проекта. В моем случае это Django, указываем название своего проекта и используемое окружение.

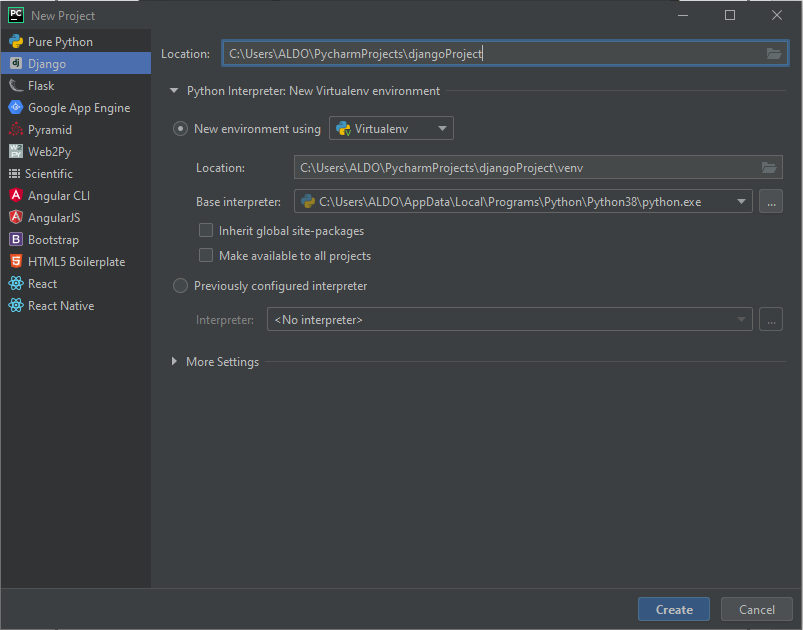


Рисунок 1.9 – Создание проекта

1.9 Интерфейс и встроенные функции Pycharm

После открытия проекта в Pycharm, у нас появится возможность начать работу.

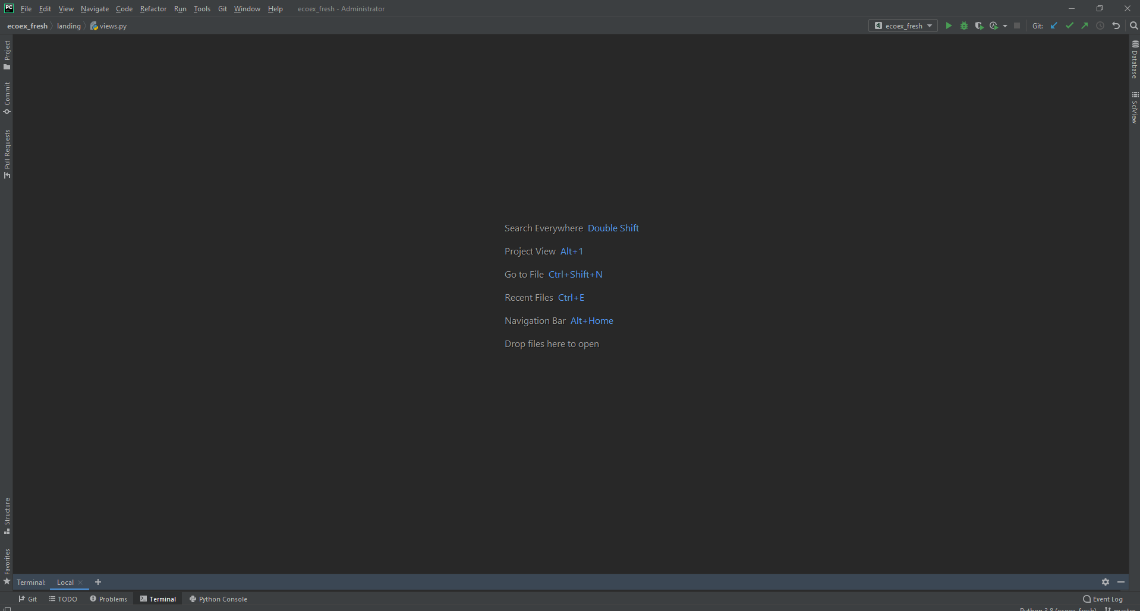


Рисунок 1.10 – Интерфейс Pycharm

Слева, во вкладке Project предоставлены все файлы внутри папки проекта, некоторые из них отмечены другим цветом и сам знак папки может немного различаться, все это сделано для удобства разработчика, чтобы последний понимал, какие файлы, хранятся в этих папках.

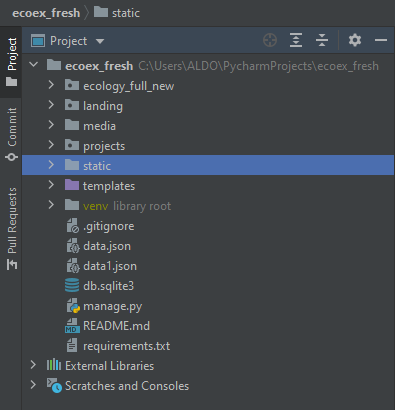


Рисунок 1.11 – Раздел Project

Сверху расположена панель со всевозможными функциями для редактирования, просмотра, поиска, рефакторинга, запуска программного кода.



Рисунок 1.12 – Верхняя панель

В Pycharm есть встроенный функционал для работы с Git. Даже не зная команд, начинающий разработчик может участвовать в командной разработке программного обеспечения. Также, очень удобно смотреть историю коммитов не выходя из приложения (см. рис. 1.13).

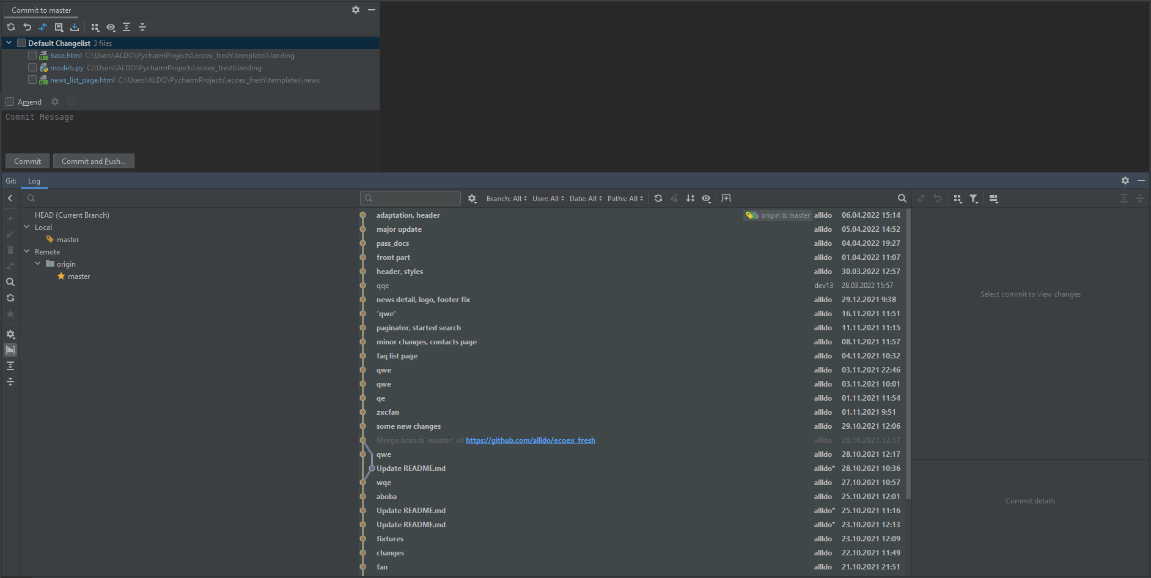


Рисунок 1.13 – Верхняя панель

Все же несмотря на то, что разработчики должны знать команды гита, я нахожу этот встроенный функционал полезным.

1.10 Виртуальное окружение

Язык программирования Python, как и большинство современных языков программирования, имеет свой уникальный способ загрузки, хранения пакетов. Однажды это приводит к вопросу где они хранятся.

Рассмотрим ситуацию, в которой есть два проекта, работающих с одним модулем, имея эту зависимость, оба проекта привязаны к этому модулю. Проблема проявляется, когда проекты запрашивают разные версии модуля и соответственно, один проект начинает работать некорректно.

Именно для таких ситуаций приходит создание изолированной среды для проектов Python. Установим пакет с изолированной средой.

Для этого достаточно в папке с проектом прописать команду.

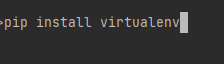


Рисунок 1.14 – Команда установки среды

Теперь, после установки мы можем создать виртуальное окружение. В зависимости от версии Python, команда различается, в своем примере я использую Python 3 (см. рис. 1.11).



Рисунок 1.15 – Команда создания изолированной среды

После этой команды начнется создание виртуального окружения, которое создаст каталог venv.

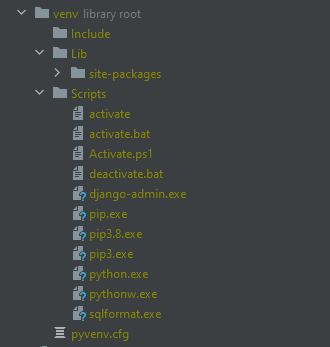


Рисунок 1.16 – Структура виртуального окружения

В новой папке есть 3 других папки:

- Include, в нее входят компилирующие пакеты Python.

- Lib, копия версии Python вместе со всеми зависимостями для проекта.

- Scripts, файлы, взаимодействующие с виртуальной средой.

Таким образом, достигается полная изоляция от глобальной среды. Поскольку виртуальное окружение было просто создано, для работы с ним необходимо его активировать (см. рис. 1.13).



Рисунок 1.17 – Активация виртуального окружения

Теперь, в случае необходимости установки пакета, можно устанавливать, не думая о том, что возникнут конфликты с другими проектами. Именно поэтому, виртуальное окружение является неотъемлемой частью навыков программистов на Python, ведь часто бывает что приходится работать со множеством проектов.

1.11 Проектирование сайта

Проектирование - этот, возможно, ключевой этап создания интернет-сайта, отвечает нам на следующие вопросы:

- Каковы наши цели — зачем мы делаем сайт?

-Как мы реализуем поставленные цели?

- Как сайт будет выглядеть и работать?

Проектирование даёт сайту очень много:

Сильно повышает гарантию достижения результата. Только четко сформулировав задачи, определив целевую аудиторию сайта и её потребности, смоделировав взаимодействие сайта и его пользователей, мы можем быть уверены - мы получим то, что нужно.

Экономит время и деньги. Ошибку на этапе проектирования довольно просто: меняем несколько кусков текста и схем. Сделать это на этапе разработки дизайна или вёрстки будет уже дороже. Если ошибка обнаруживается на этапе программирования, её исправление может стоить многие тысячи (десятки, сотни тысяч) рублей и занять месяцы, а то и годы.

Позволяет эффективно разделять работу. Проектное задание — это вполне самодостаточный документ. Получив его, клиент может сделать сайт своими силами или нанять другую команду, которая, по его мнению, лучше справится с непосредственно разработкой.

Проектирование можно условно разбить на четыре основные части:

- Целеполагание.

- Исследование контекста,

- Создание концепции.

- Моделирование.

Целеполагание необходимо для определения того, зачем мы делаем сайт и каких именно результатов хотим достичь. Это служит ориентиром для всей дальнейшей работы: что бы мы ни делали - будь то моделирование, создание интерфейсов, добавление новых функций или изменение старых - всё это должно соответствовать целям. В будущем они же помогают оценить успешность проекта. Первый шаг в целеполагании сделан в видении проекта. На этапе же проектирования цели формулируются более точно и подробно, а также определяются задачи сайта, выполнение которых будет работать на достижение каждой цели.

Исследование необходимо для получения информации, которую мы будем называть контекстом сайта. Под контекстом мы понимаем различные обстоятельства, окружающие сайт и способные оказать влияние на его работу. К таким обстоятельствам относятся:

- Целевая аудитория и её потребности.

- Характеристика и тенденции области.

- Конкуренты и их деятельность.

- Опыт других проектов.

- Законодательные или иные ограничения.

Контекст проекта помогает нам понять целевую аудиторию и то, каким нужно сделать сайт: как его позиционировать, какая информация на нём должна быть и на каком языке он должен говорить с ЦА (это называют коммуникативной стратегией), как он будет отличаться от конкурентов (естественно, в выгодную сторону).

Кроме того, исследование погружает команду проекта в тему, позволяет где-то даже на подсознательном уровне видеть/принимать правильные решения.

Лучше всего, конечно, исследование помогает понять аудиторию:

Как с ней общаться, какие слова использовать в «разговоре» с ней, как называть свой продукт или услугу (возможно, нужно использовать профессиональный сленг, или наоборот, нарочито упрощённую терминологию), обращаться ли к ней на «вы» или на «ты». Каковы её потребности: что приоритетно, что необязательно, но желаемо, что нельзя делать ни в коем случае.

В идеале, данные о контексте мы должны получить от клиента, но если таких данных у клиента нет, то исследование контекста предстоит провести самим.

Для исследования советуется использовать два метода, которые лучше всего зарекомендовали себя на всех без исключения проектах, которые мы делали:

- Исследование доступных источников — литература, интернет-ресурсы и так далее.

- Интервью с ключевыми действующими лицами — пользователями, экспертами.

Без данных о контексте шансы сделать хорошо пользователю стремятся к нулю. Даже самостоятельное исследование даёт полезную информацию, пусть и в меньшем объёме, нежели профессиональное.

И даже если вы настолько неумело проводите исследование (что маловероятно), что не можете получить ответы на выше обозначенные вопросы, вы всё равно погружаетесь в контекст проекта и, хотя бы подсознательно улавливаете что-то полезное.

После поставления цели и получения контекста пора облачить всю имеющуюся информацию в концепцию (аналог того, что мы сделали в видении, но более подробную). Под концепцией мы понимаем основные идеи и возможности, заложенные в проект:

- Что и для кого мы делаем — общая идея и целевая аудитория.

- Как сайт будет работать и какую информацию содержать.

- Как сайт будет зарабатывать (если это проект с прямой монетизацией).

- Каковы будут отличительные особенности сайта (от конкурентов), как он будет позиционироваться.

- Как сайт будет развиваться после запуска.

- Концепция задаёт направление проектированию и помогает, аналогично видению, ещё раз состыковать точки зрения на проект - нашу и клиента.

Моделирование — это создание модели сайта, которая описывает функциональные возможности и информационную структуру. В функциональной части модели мы описываем возможности, которые сайт предоставляет своим пользователям: например, выкладывать, группировать и комментировать фотографии (социальная сеть) или заказывать и оплачивать товар (интернет-магазин). Важно понимать, что возможности являются инструментами решения задач. Если придуманная возможность не решает ни одну из задач — это может означать, что она лишняя. В проектном задании мы описываем возможности на достаточно высоком уровне абстракции — не так детально, как мы сделаем это в «техническом задании на программирование», поскольку здесь это просто не требуется.

Информационная структура — это схема, показывающая, из каких разделов состоит сайт, какие задачи они решают, и как пользователь будет перемещаться по сайту (схема навигации).

Далее, прорабатываем схему раздела — это более глубоко и детально проработанная схема (по сравнению с информационной структурой сайта), показывающая навигацию по разделу, связи и переходы между подразделами. Схема раздела в идеале включает следующие элементы:

- Задачи. Какие из ранее поставленных задач решает раздел. Например, раздел «Фотографии» в социальной сети решает задачу обмена информацией между друзьями и последующего общения.

- Сообщения, в буквальном смысле сообщения, которые раздел или его часть передаёт посетителю. Например, сообщение приветствия «Привет, ты на портале! Мы рады видеть тебя здесь» или «Эй, это наш лучший товар — попробуй, закажи сейчас и убедись в этом сам!».

Сообщения бывают разных типов; наиболее часто встречающиеся: рекламные, призывы к действию, уведомления и имиджевые сообщения.

Функциональные элементы — элементы интерфейса, дающие возможность посетителю выполнить какую-то операцию. Например, функциональным элементом является форма для ввода сообщения, позволяющая отправить сообщение, или кнопка в интерфейсе, сохраняющая сделанные изменения.

Варианты поведения посетителя — предположения о том, что посетитель сайта может или должен сделать после изучения интерфейса или отдельных его частей.

1.12 Верстка методом Mobile First

Свой проект я решил сделать, используя принцип Mobile First. Обычно создание сайта выглядит так: дизайнер создает дизайн для ПК версии, далее для мобильной потом программист собираем сайт.

Mobile First подразумевает изначальное создание именно мобильной версии. То есть первым делом веб-разработчики идеально подгоняют проект под смартфоны и лишь потом адаптируют его для отображения на десктопах. При таком подходе не возникает ощущение, что мобильная версия обрезана или недоработана – элементы не наезжают друг на друга, а все меню и ссылки корректно работают.

Первые шаги к Mobile First были сделаны в 2015 году. Google внедрила новый алгоритм ранжирования, согласно которому сайты без мобильной версии потеряли позиции в поисковой выдаче.

На конференции SMX Advanced в 2017 году компания пошла еще дальше и запустила индексацию по принципу Mobile First. Поисковик начал рассматривать мобильные версии сайтов как основные, а качество их оптимизации учитывать при формировании выдачи.

Полный переход на Mobile First займет несколько лет, но откладывать адаптацию сайта под данный метод не стоит.

Если посмотреть статистику своего сайта в Google Analytics, можно заметить, что 50% трафика или даже больше приходится на мобильные устройства. С каждым годом этот показатель будет только расти.

В 2017 году было проведено несколько масштабных исследований на тему мобильных сайтов, вот самые интересные данные:

- 60% опрошенных заявили, что вряд ли вернутся на сайт с неудобной мобильной версией;

- 57% пользователей уверены, что никогда не будут рекомендовать интернет-бизнес с плохим мобильным сайтом;

- 80% сайтов из рейтинга Alexa Rank имеют адаптированные мобильные версии;

- 69% времени в сети приходится на использование смартфонов;

- Каждый час с мобильных устройств осуществляется 68 млн поисковых запросов в Google;

- 35% мобильных поисковых запросов совершается с устройств с 5,1-дюймовыми экранами. Самое популярное разрешение дисплея – 720 x 1280 точек.

Сайт в адаптивном дизайне состоит из нескольких макетов. В каждом из них – структура размещения элементов на сайте под разные размеры экранов. Движок определяет диагональ, разрешение и соотношение сторон дисплея, на котором пользователь открыл страницу, и загружает соответствующий шаблон.

Например, для устройств с экранами в диапазоне от четырех до шести дюймов (смартфоны) создается один макет, от семи до одиннадцати (планшеты) – второй, а от двенадцати и больше (ноутбуки) – третий.

В отзывчивом дизайне макет сайта подстраивается под ширину экрана, на котором человек смотрит страницу. Даже если окно браузера изменится на один пиксель, страница автоматически адаптируется под новые размеры.

Такой подход позволяет не тратить время на создание мобильной версии сайта. При этом будет не лишним убедиться, что все элементы нормально смотрятся на разных устройствах.

Адаптивный и отзывчивый дизайны похожи – оба они подстраиваются под размеры экранов. Разница в том, что в первом случае контент может быть только фиксированных размеров, предусмотренных макетом, а во втором – способен динамически меняться.

Проблема адаптивного и отзывчивого дизайнов в том, что большинство веб-разработчиков, используя эти методики, сначала создают сайт под компьютеры и уже потом адаптируют его под мобильные устройства. Как результат – на смартфонах страницы выглядят хуже.

Стратегия Mobile First строится от обратного: сначала сайт делается под смартфоны, а затем дорабатывается для десктопов. В этом случае нет ощущения, что мобильная версия неполноценна.

Хороший пример – сайт Esquire. Он создан по принципу Mobile First:

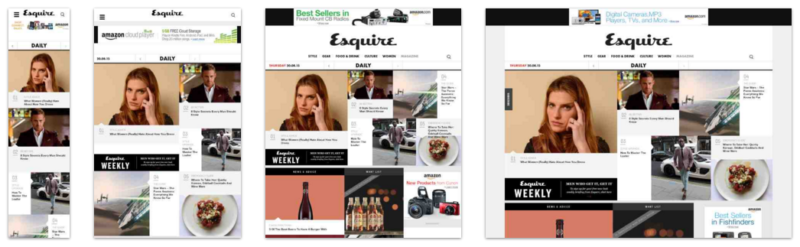


Рисунок 1.18 – Пример сайта

Чек-лист: каким должен быть сайт Mobile First.

- Информация отображается корректно как в книжной, так и в альбомной ориентации;

- Все кликабельные элементы работают;

- Отсутствуют тяжелые изображения, которые замедляют загрузку страницы;

- На сайте нет ничего, связанного с Flash;

- Нет всплывающих окон, которые перекрывают полстраницы;

- Нигде не задействована горизонтальная прокрутка;

- Контактный номер кликабельный или его хотя бы можно скопировать;

- Поля в форме регистрации короткие и простые, отключена функция автозаполнения (часто может подтягиваться некорректная информация);

- Все кнопки сделаны с помощью CSS, нет кнопок-изображений;

- Размер картинок меняется автоматически в зависимости от типа устройства;

- Есть возможность совершить покупку в один клик.

Общие черты этих сайтов: дизайн не перегружен, отчетливо видны поисковые строки и корзины, есть кликабельные номера, а также вспомогательные элементы, которые подводят пользователя к покупке.

1.13 Верстка проекта

Верстка — это описание визуальной части сайта с помощью гипертекстового документа на основе HTML-разметки. Проще говоря, соединение и расположение на странице документа разных элементов веб-сайта: текстовых блоков, изображений, таблиц, видео и т.д.

Верстка сайта — один из наиболее важных этапов создания интернет-ресурса. От нее зависит:

- Скорость загрузки сайта.

- Корректность его отображения в браузере.

- Соответствие стандартам HTML и требованиям поисковых систем.

- Адаптивность под разные устройства пользователей (планшет, смартфон, ноутбук).

Верстка относится к сфере frontend-разработки. Она помогает правильно расположить элементы на странице и сделать так, чтобы работать с ними было максимально удобно.

HTML - язык разметки, при помощи которого браузеры показывают нам размер, порядок, шрифт и форму текста.

Тег <div> — блочный элемент и обычно используется для группировки крупных элементов. Верстальщик создает в интегрированной среде разработки IDE макет сайта: готовит фоны и изображения, размечает, где какая область (верх, низ, основной контент, боковые панели) будет находиться. Каждая часть страницы размещается в соответствующем блоке <div>.

Все стилевые решения вынесены за границы HTML-кода в каскадные таблицы стилей (CSS). С ними можно менять цвет, размеры, расположение, границы и другие свойства блоков <div>. В HTML-документе это прописывается так: <div class=”one”></div>, фоновый цвет задается следующим образом: background: red. В итоге получается красный блок. А если нужно ограничить его размер 250 пикселями, то необходимо прописать: width: 250px и height: 250px.

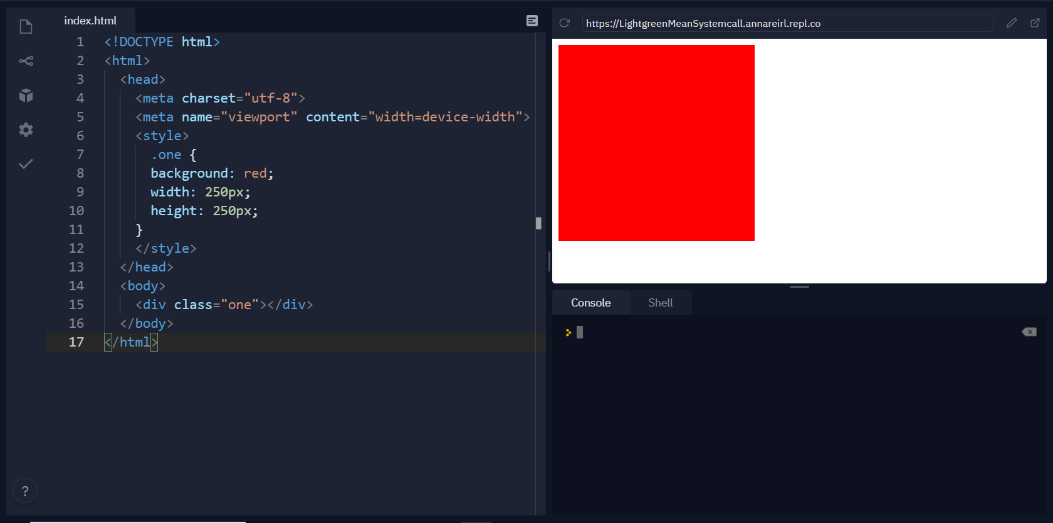


Рисунок 1.19 – Пример кода HTML

В программировании валидность означает соответствие кода стандарту W3C (Консорциума Всемирной паутины) и корректное отображение веб-сайта на разных устройствах. Этот показатель обеспечивается соблюдением правил построения кода, его оптимизации и минимизации. Благодаря такой верстке сайт может занять более высокие позиции в поисковой выдаче.

Чтобы верстка была валидной, нужно как минимум, чтобы элементы HTML-кода содержались в соответствующих разделах.

После получения макета представляется будущая страницу в виде HTML-разметки, а после начинается написание кода. Это поможет избежать неправильной вложенности тегов и некоторых других ошибок.

Метод [слепой печати](https://www.ratatype.ru/learn/) поможет научиться быстро набирать код, не отвлекаясь от макета и внешнего вида веб-сайта в браузере.

Писать HTML-код можно в любом редакторе кода, даже в «Блокноте», но это долго и неэффективно. Верстальщик может установить IDE (интегрированную единую среду разработки, которая используется разработчиками для создания проекта). От «Блокнота» она отличается массой возможностей: наличием встроенного проводника, подсветкой синтаксиса, управлением форматированием кода, возможностью работы с Git (инструментом, позволяющим фиксировать изменения в коде) и пр.

С помощью БЭМ можно создавать расширяемые и повторно используемые компоненты интерфейса. В результате вам больше не потребуется тратить время на придумывание названий классов или того, как элементы будут зависеть друг от друга.

Препроцессор Sass помогает в организации кода, избавляет от написания селектора каждый раз с новой строки, позволяет сократить код и вынести одинаковые реализации.

Сборщики проекта выполнят то, что разработчику не хочется делать руками и что отнимает много времени. Например, Gulp позволяет:

- Автоматически обновлять страницу браузера при сохранении кода.

- Расставлять префиксы для поддержки разных браузеров.

- Автоматически собирать все файлы в один.

- Создавать SVG-спрайты.

Признаки правильной верстки:

- Структура страницы составлена в HTML, параметры стилей присвоены в отдельном CSS.

- Использовано небольшое количество изображений. Все по максимуму прописано текстом или кодом.

- Верстка корректно отображается при разных разрешениях во всех популярных браузерах: Chrome, Opera, Firefox, Safari.

- Атрибуты элементов <title>, <alt>, <description> корректно заполнены.

- Использованы заголовки разных уровней (h1, h2, …, h6).

Инструменты верстальщика:

- Программы для тестирования: Markup Validator, CrossBrowserTesting

- Программы для работы с изображениями: Photoshop, Krita, Gimp и пр.

- Редакторы кода: VS Code, Webstorm, Notepad++, SublimeText, Webstorm.

Процесс базового тестирования предполагает ручные прогоны сценариев в разных браузерах и на разных устройствах. В этих сервисах можно найти точную информацию: validator.w3.org и jigsaw.w3.org/css-validator. Также нужно проверить верстку на соответствие макету с помощью сервиса welldonecode.com/perfectpixel. Page Ruler поможет измерить в пикселях сверстанные блоки, оценить размеры форм, шапок, логотипов, контактных данных, карточек, рубрик, виджетов и пр.

1.14 MVC во фреймворке Django

Django - это фреймворк для создания веб-приложений с помощью языка программирования Python. Django был создан в 2005 году, когда веб-разработчики из газеты Lawrence Journal-World стали использовать Python в качестве языка для создания веб-сайтов. А в 2008 году вышел публичный первый релиз фреймворка. На сегодняшний день он продолжает развиваться. Так, текущей версией фреймворка на момент написания этой статьи является версия 4.0, которая вышла в декабре 2021 года. Ну и также постоянно выходят подверсии. Django довольно популярен. Он используется на многих сайтах, в том числе таких, как Pinterest, PBS, Instagram, BitBucket, Washington Times, Mozilla и многих других. Фреймворк является бесплатным. Он развивается как open source, его исходный код открыт, его можно найти репозитории на githube. На Django можно создавать широкий диапазон веб-приложений: от небольших персональных сайтов до высоконагруженных сложных веб-сервисов. Django по умолчанию предлагает готовую функциональность для ряда распространенных задач, например, систему аутентификации, генерацию карт сайта и т.д., благодаря чему нам можно не изобретать велосипед и достаточно взять уже готовые компоненты. В Django большое внимание уделяется безопасности, благодаря чему фреймворк помогает разработчикам избежать многих распространенных проблем в системе безопасности, например, sql-инъекций. Фреймворк Django реализует архитектурный паттерн Model-View-Template или сокращенно MVT, который по факту является модификацией распростаненного в веб-программировании паттерна MVC (Model=-View-Controller). Схематично мы можем представить архитектуру MVT в Django следующим образом:

- URL dispatcher: при получении запроса на основании запрошенного адреса URL определяет, какой ресурс должен обрабатывать данный запрос.

- View: получает запрос, обрабатывает его и отправляет в ответ пользователю некоторый ответ. Если для обработки запроса необходимо обращение к модели и базе данных, то View взаимодействует с ними. Для создания ответа может применять Template или шаблоны. В архитектуре MVC этому компоненту соответствуют контроллеры (но не представления).

- Model: описывает данные, используемые в приложении. Отдельные классы, как правило, соответствуют таблицам в базе данных.

- Template: представляет логику представления в виде сгенерированной разметки html. В MVC этому компоненту соответствует View, то есть представления.

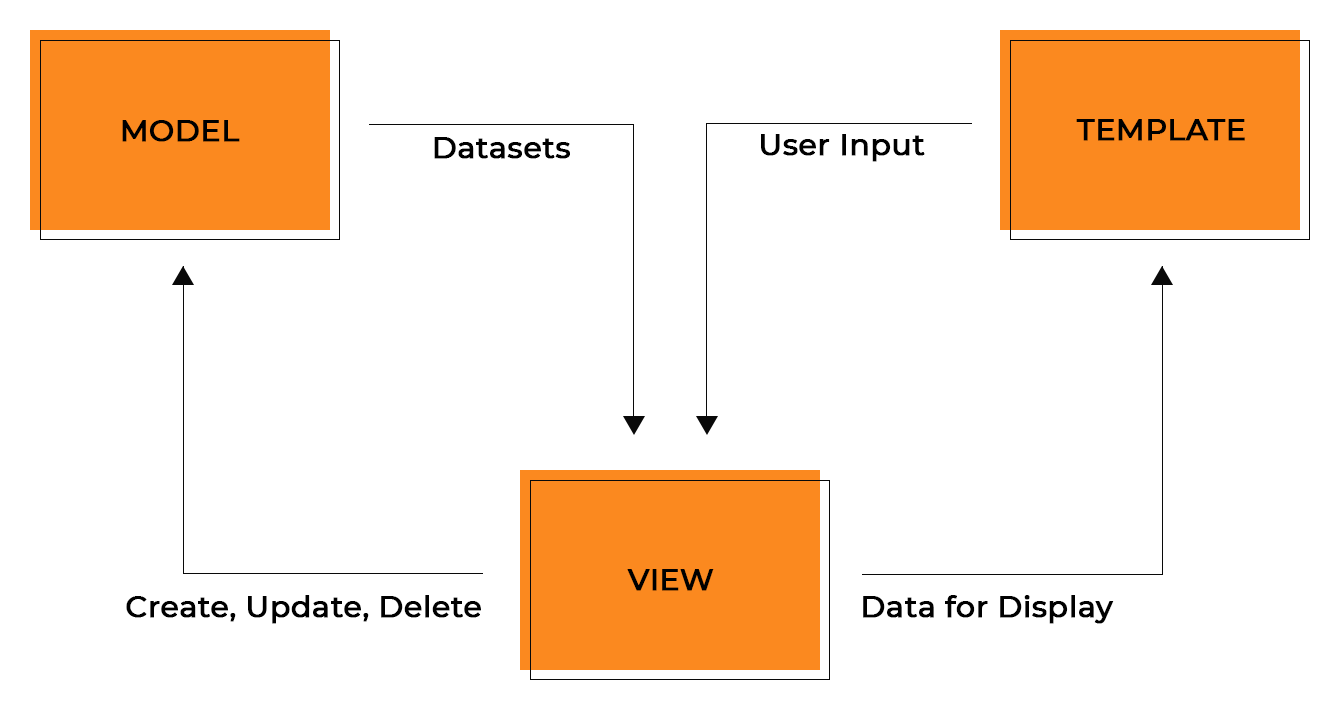


Рисунок 1.20 – Визуализация паттерна

Когда к приложению приходит запрос, то URL dispatcher определяет, с каким ресурсом сопоставляется данный запрос и передает этот запрос выбранному ресурсу. Ресурс фактически представляет функцию или View, который получает запрос и определенным образом обрабатывает его. В процессе обработки View может обращаться к моделям и базе данных, получать из нее данные, или, наоборот, сохранять в нее данные. Результат обработки запроса отправляется обратно, и этот результат пользователь видит в своем браузере. Как правило, результат обработки запроса представляет сгенерированный html-код, для генерации которого применяются шаблоны.

2 ЭКОНОМИЧЕСКИЙ РАЗДЕЛ

2.1 Введение

Веб-сайты в настоящее время полностью интегрированы в человеческую жизнь, деятельность человека в различных областях, науку, поэтому необходимо обеспечить их различными полезными сервисами. Причиной быстрого развития информационных технологий является прямой вклад в рост потребности в быстрой и качественной обработке данных, что способствует развитию общества.

Тема дипломного проекта: Разработка web-сайта для ТОО "Ecology Expert”.

Экономическая часть моей дипломной работы рассчитывает стоимость создания веб-сайта, себестоимость проекта, рентабельность проекта, а также срок окупаемости.

2.2 Расчёт стоимости организации мобильного приложения

2.2.1 Расходы на заработную плату

Фонд оплаты труда состоит из основной и дополнительной заработной платы. Формула:

(2.1)

Для расчета основной заработной платы необходимы данные по трудоемкости человеко–дней, человеко–месяцев, численности и размер установленного оклада за месяц. Длительность цикла в днях по каждому виду работы укрупнено можно определить по формуле:

(2.2)

где Т – трудоёмкость этапа, нормо–час;

qn – количество исполнителей по этапу;

7 – продолжительность рабочего дня, час;

К – коэффициент выполнения норм времени (К=1,1).

В таблице 2.1 сведения о разработчиках, нормах, трудоёмкости и соответственно длительности поэтапного цикла.

Таблица – 2.1 План организации веб-сайта

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Наименование этапов | Исполнители | Трудоёмкость | | Количество исполнителей | Длительность цикла, дней |
| Нормы Часы | % от общей трудоёмкости |
| Предпроектное исследование | Разработчик | 150 | 20% | 1 | 19 |
| Техническое задание | Разработчик | 465 | 20% | 3 | 20 |
| Помощник разработчика |
| Руководитель |
| Подготовка и внедрение устройств и технологий | Разработчик | 460 | 35% | 2 | 30 |
| Помощник разработчика |
| Тестирование | Разработчик | 150 | 5% | 2 | 10 |
| Помощник разработчика |
| Сдача проекта | РАзработчик | 100 | 5% | 2 | 6 |
| Руководитель |

Затраты на заработную плату разработчиков рассчитываются по следующей формуле:

(2.3)

где Тpi – трудоемкость работ i–го разработчика, чел.мес;

Зi – основная заработная плата i–го разработчика (оклад), тг/мес.

Учитывая, что в месяце 24 рабочих дня, то ежедневная трудоемкость составит 1/24 ежемесячной трудоемкости.

Оклад администратора составляет 80 000 тенге в месяц. Оклад помощника администратора составляет 60 000 тенге. Оклад руководителя проекта составляют 100 000 тенге в месяц.

Исходя из вышеперечисленных данных поэтапно посчитаем затраты на заработную плату для каждого участника проекта.

Предпроектное исследование.

Для разработчика: Зp= 19/2480000=63 333 тенге.

Техническое задание.

Для разработчика: Зp= 20/2480000=66 666 тенге.

Для помощника разработчика: Зp= 20/2460000=50 000 тенге.

Для руководителя проекта: Зp= 20/24100000=83 333 тенге.

Подготовка и внедрение устройств и технологий.

Для разработчика: Зp= 30/2480000=100 000 тенге.

Для помощника разработчика: Зp= 30/2460000=75 000 тенге.

Тестирование.

Для разработчика: Зp= 10/2480000=33 333 тенге.

Для помощника разработчика: Зp= 10/2460000=25 000 тенге.

Сдача проекта.

Для разработчика: Зp= 6/2480000=20000 тенге.

Для руководителя проекта: Зp= 6/24100000=25 000 тенге.

Результаты расчета затрат по заработной плате с учетом трудоемкости и установленных окладов каждого участника проекта представлены в таблице 2.2. Также основная заработная плата рассчитывается как сумма оплаты труда всех работников, задействованных в разработке, имеется в таблице 2.2.

Таблица – 2.2 Основная заработная плата разработчиков

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Наименование  этапа | Исполнители | Трудоёмкость | | Оклад, тг./мес. | Затраты по з/п, тг |
| чел. дн | чел. мес. |
| 1 | Предпроектное исследование | Разработчик | 19 | 0,791 | 80000 | 63333 |
| 2 | Техническое задание | Разработчик | 20 | 0,833 | 80000 | 66666 |
| Помощник разработчика | 60000 | 50000 |
| Руководитель | 100000 | 83333 |
| 3 | Подготовка и внедрение устройств и технологий | Разработчик | 30 | 1,25 | 80000 | 100000 |
| Помощник разработчика | 60000 | 75000 |
| 4 | Тестирование | Разработчик | 10 | 0,416 | 80000 | 33333 |
| Помощник разработчика | 60000 | 25000 |
| 5 | Сдача проекта | Разработчик | 6 | 0,25 | 80000 | 20000 |
| Руководитель | 100000 | 25000 |
| Итого: 541 665,00 | | | | | | |

Дополнительная заработная плата (премии и т.д.) разработчиков составляет 20 % от основной заработной платы и рассчитывается по формуле:

(2.4)

Согласно формуле (2.4) дополнительная заработная плата будет равна:

Здоп=5416650,2/1=108 333 тенге.

Тогда фонд оплаты труда в соответствии с формулой (2.1) составит:

ФОТ=541665+108333=649998 тенге.

Социальный налог согласно ст. 358 НК РК с 01.01.09 составляет 11% от дохода работника (ЗП + дополнительная оплата труда).

(2.5)

В соответствии со статьями налогового кодекса 144 и 145 о доходах, не подлежащих налогообложению, отчисления в пенсионный фонд в размере 10% от фонда оплаты труда социальным налогом не облагаются. Пенсионные отчисления рассчитываются по формуле:

(2.6)

Согласно формуле (2.6) пенсионные отчисления будут равны:

ПО=6499980,1/1=64999,8 тенге.

Тогда отчисления по социальному налогу согласно формуле (2.5) составят:

Ос= (649998–64999,8)0,11=70784,8 тенге.

Прочие расходы (Пр) включают расходы на машинное время (порядка 3–х месяцев на подготовку, внедрение и тестирование устройство: 700 часов стоимостью 10 тг. /час).

Пр=70010=7000 тенге.

2.2.2 Расчет затрат на оборудование и его обслуживание

Для организации и разработки мобильного приложения по дипломному проекту необходимы устройства, которые указаны в таблице 2.3.

Таблица – 2.3 Расчет оборудования для локальной сети

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование оборудования | Количество,  штук | Цена за единицу в тенге | Сумма в тенге |
| Персональный компьютер | 2 | 150 000 | 300 000 |
| Интернет соединение | 1 | 5 000 | 5 000 |
| Итого: | | | 305 000 |

Общая стоимость расходов на оборудование составляет 305000 тенге.

2.2.3 Амортизационные отчисления

Расчет затрат на амортизацию производится по формуле:

(2.7)

A – ежегодная сумма амортизационных отчислений;

СПЕР – первоначальная стоимость объекта;

НА – норма амортизационных отчислений.

В соответствие со ст. 20 Налогового кодекса РК с 01.01.09, норма амортизационных отчислений (НА) на оборудование связи составляет 25%.

А=0,25305000/0,1=76 250 тенге.

Амортизация на устройства защищенной локальной сети согласно формуле (2.7) будет равна 329 247,5 тенге.

А=76 250/1224=264 тенге.

Дневная сумма амортизации на устройства локальной сети составит 1143 тенге.

Так как длительность непосредственного внедрения проекта ииспользования данного оборудования составляет 85 дней, то общие затраты наамортизацию будут равны 22440тенге.

А=26485=22 440тенге.

2.2.4 Затраты на электроэнергию

Важной статьей затрат являются затраты на потребляемую электроэнергию. Затраты на электроэнергию рассчитывается по следующей формуле:

(2.8)

где W – потребляемая мощность, кВт;

Т – количество часов работы;

S – стоимость киловатт–часа электроэнергии.

Виды используемого оборудования, а также потребляемая ими мощность представлены в таблице 2.5. Исходя из этих данных рассчитывается стоимость расхода электроэнергии.

Согласно установленному тарифу по энергопотреблению стоимость 1 кВт составляет 8,84 тенге.

С учетом 24–часовой непрерывной работы оборудования и длительности разработки, внедрения проекта, количество часов работы составит:

Т=2485=2040 часов.

Таблица – 2.5 Потребляемая мощность оборудования

|  |  |  |
| --- | --- | --- |
| Наименование оборудования | Количество,  штук | Потребляемая мощность,  кВт/час |
| Персональный компьютер | 2 | 2,4 |
| Маршрутизатор | 1 | 0,03 |
| Итого: | | 2,43 |

В соответствии с формулой (2.8) расходы на электроэнергию составят

Сэл=2,438,842040=43 821,648 тенге.

2.2.5 Расчет на накладные расходы

Накладные расходы на разработку проекта составляют 10% от общей суммы затрат и рассчитываются по формуле:

(2.9)

Тогда, согласно формуле (2.9), накладные расходы будут равны 114 204,4 тенге.

Нр=(649998+70784,8+305000+22440+43821,648+7000)0,1=114 204,4 тенге.

2.3 Расчет себестоимости проекта

Расчет осуществляется по калькуляционным статьям расходов. Прежде всего, надо рассчитать себестоимость на протяжении всего жизненного цикла. Себестоимость — это все затраты на производство и реализацию проекта.

Себестоимость организации защищенной локальной сети складывается из следующих статьей затрат:

* заработная плата основных разработчиков;
* дополнительная заработная плата;
* фонд оплаты труда;
* отчисления с фонда оплаты труда в социальный налог;
* амортизационные отчисления;
* расходные материалы;
* накладные расходы.

Таким образом, себестоимость разработки проекта определяется по следующей формуле:

(2.10)

где ФОТ – фонд оплаты труда (основная и дополнительная заработная плата);

Ос – социальный налог;

Pо – расходы на оборудование и обслуживание;

А – амортизационные отчисления;

Сэл – расходы на электроэнергию;

Нр – накладные расходы;

Пр – расходы на машинное время.

В соответствии с произведенными расчетами по статьям затрат себестоимость проекта, согласно формуле (2.10), будет равна 1 142 044,448 тенге.

С=649998+70784,8+305000+22440+43821,648+114204,4 +7000=1 213 248,660 тенге.

2.4 Рентабельность проекта

Для того чтобы рассчитать рентабельность проекта, нужно найти прибыль. Прибыль рассчитывается по формуле:

(2.11)

где Ур=25% – норматив рентабельности, составляющий прибыль учреждения;

П–прибыль.

П=0,25\*1 213 248,660 = 303312,165 тенге.

Рентабельность проекта рассчитывается по формуле:

(2.12)

Р= 303312,165 /1 213 248,660 1=0,25=25%.

Рентабельность проекта равна 25%.

2.5 Срок окупаемости проекта

В качестве условного годового дохода, полученного после организации веб-сайта составит, к примеру 1000000 тенге.

Срок окупаемости рассчитаем по формуле:

(2.14)

где Ток – срок окупаемости.

Ток=1 213 248,660/303312,165 =1,8>1год

2.6 Заключение

Проведенные исследования показали, что данная дипломная работа является экономически выгодной и актуальной на сегодняшний день. Веб-сайт для компаний является неотъемлимой частью ведения бизнеса. Приведенный расчет показал, что общие затраты на разработку веб-сайта с учетом налогов составляют 1 213 248,660тенге.

Проект окупается меньше чем за год, в последующие годы ПО принесет только прибыль. Срок окупаемости является оптимальным с учетом инфляции и необходимыми затратами, производимыми на выплату нормального функционирования всего проекта и отдельных его частей.

В целом дипломный проект на тему: «Разработка веб-сайта для ТОО “Ecology Expert”», является актуальной и востребованной темой, для данной компании.

3 ОХРАНА ТРУДА

Научно–технический прогресс внес серьезные изменения в условия производствен¬ной деятельности работников умственного труда. Их труд стал более интенсивным, напря¬женным, требующим значительных затрат умственной, эмоциональной и физи¬ческой энергии. Это потребовало комплексного решения проблем эргономики, ги¬ги¬ены и ор-ганизации труда, регламентации режимов труда и отдыха.

Специалист по защите информации работает в условиях интенсивного интеллектуального умственного труда в офисном помещении. У него малоподвижный образ жизни, возможен ненормированный рабочий день. Основные средства труда специалиста по информационной безопасности – это разнообразные средства вычислительной техники, сетевых технологий, собственная база знаний и опыта. При работе с ком¬пьютером человек подвергается воздействию ряда опасных и вредных производственных факторов: электромагнитных полей, инфракрасного и ионизирующего излучений, шума и виб¬рации, статического электричества и других

В данном разделе дипломного проекта рассмотрены следующие вопросы:

 безопасность труда и ее общие положения;

 производственная санитария;

 техника безопасности, в частности, электробезопасность и пожарная безопасность на предприятии.

3.1 Безопасность труда

3.1.1 Общие положения труда техника по защите информации

К работе с ПК допускаются работники, не имеющие медицинских противопоказаний, прошедшие инструктаж по вопросам охраны труда, с группой по электробезопасности не ниже I.

Женщины со времени установления беременности и в период кормления грудью к выполнению всех видов работ, связанных с использованием ПК, не допускаются. Лица, не моложе 18 лет, прошедшие обязательный при приеме на работу и ежегодные медицинские освидетельствования на предмет пригодности для работы на ЭВМ и ПЭВМ.

Прошедшие вводный инструктаж по охране труда; прошедшие инструктаж по охране труда на конкретном рабочем месте и знающие правила работы на персональном компьютере, о чем должна быть сделана соответствующая запись в журнале; – прошедшие обучение работе на персональном компьютере с использованием конкретного программного обеспечения

При работе с ПК на работников могут оказывать неблагоприятное воздействие следующие опасные и вредные производственные факторы:

 повышенный уровень электромагнитных излучений;

 повышенный уровень ионизирующих излучений;

 повышенный уровень статического электричества;

 повышенная напряженность электростатического поля;

 повышенная или пониженная ионизация воздуха;

 повышенная яркость света;

 прямая и отраженная блесткость;

 повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека;

 статические перегрузки костно–мышечного аппарата и динамические локальные перегрузки мышц кистей рук;

 перенапряжение зрительного анализатора;

 умственное перенапряжение;

 эмоциональные перегрузки;

 монотонность труда.

В зависимости от условий труда, в которых применяются ПК, и характера работы на работников могут воздействовать также другие опасные и вредные производственные факторы.

Организация рабочего места с ПК должна учитывать требования безопасности, удобство положения, движений и действий работника.

Рабочий стол с учетом характера выполняемой работы должен иметь достаточный размер для рационального размещения монитора (дисплея), клавиатуры, другого используемого оборудования и документов, поверхность, обладающую низкой отражающей способностью.

Клавиатура располагается на поверхности стола таким образом, чтобы пространство перед клавиатурой было достаточным для опоры рук работника (на расстоянии не менее чем 300 мм от края, обращенного к работнику).

Чтобы обеспечивалось удобство зрительного наблюдения, быстрое и точное считывание информации, плоскость экрана монитора располагается ниже уровня глаз работника предпочтительно перпендикулярно к нормальной линии взгляда работника (нормальная линия взгляда – 15 град. вниз от горизонтали). Для исключения воздействия повышенных уровней электромагнитных излучений расстояние между экраном монитора и работником должно составлять не менее 500 мм (оптимальное 600 – 700 мм).

Применяемые подвижные подставки для документов (пюпитры) размещаются в одной плоскости и на одной высоте с экраном.

Рабочий стул (кресло) должен быть устойчивым, место сидения должно регулироваться по высоте, а спинка сиденья – по высоте, углам наклона, а также расстоянию спинки от переднего края сиденья. Регулировка каждого параметра должна быть независимой, легко осуществляемой и иметь надежную фиксацию.

Для тех, кому это удобно, предусматривается подставка для ног. Рабочее место размещается таким образом, чтобы естественный свет падал сбоку (желательно слева). Для снижения яркости в поле зрения при естественном освещении применяются регулируемые жалюзи, плотные шторы. Светильники общего и местного освещения должны создавать нормальные условия освещенности и соответствующий контраст между экраном и окружающей обстановкой с учетом вида работы и требований видимости со стороны работника. Освещенность на поверхности стола в зоне размещения рабочего документа должна составлять 300 – 500 люкс.

Возможные мешающие отражения и отблески на экране монитора и другом оборудовании устраняются путем соответствующего размещения экрана, оборудования, расположения светильников местного освещения. При рядном размещении рабочих столов расположение экранов видеомониторов навстречу друг другу из–за их взаимного отражения не допускается.

Для обеспечения безопасности работников на соседних рабочих местах расстояние между рабочими столами с мониторами (в направлении тыла поверхности одного монитора и экрана другого монитора) должно быть не менее 2,0 м, а расстояние между боковыми поверхностями мониторов – не менее 1,2 м.

Для снижения уровня напряженности электростатического поля при необходимости применяются экранные защитные фильтры. При эксплуатации защитный фильтр должен быть плотно установлен на экране монитора и заземлен.

Для обеспечения оптимальных параметров микроклимата проводятся регулярное в течение рабочего дня проветривание и ежедневная влажная уборка помещений, используются увлажнители воздуха.

При работе с ПК обеспечивается доступ работников к первичным средствам пожаротушения, аптечкам первой медицинской помощи.

Работники при работе с ПК с учетом воздействующих на них опасных и вредных производственных факторов обеспечиваются средствами индивидуальной защиты в соответствии с типовыми отраслевыми нормами для соответствующих профессий и должностей.

При работе с ПК работники обязаны:

 соблюдать режим труда и отдыха, установленный законодательством, правилами внутреннего трудового распорядка организации, трудовую дисциплину, выполнять требования охраны труда, правил личной гигиены;

 выполнять требования пожарной безопасности, знать порядок действий при пожаре, уметь применять первичные средства пожаротушения;

 курить только в специально предназначенных для курения местах;

 знать приемы оказания первой помощи при несчастных случаях на производстве;

 о неисправностях оборудования и других замечаниях по работе с ПК сообщать непосредственному руководителю или лицам, осуществляющим техническое обслуживание оборудования.

Не допускается: выполнять работу, находясь в состоянии алкогольного опьянения либо в состоянии, вызванном употреблением наркотических средств, психотропных или токсических веществ, а также распивать спиртные напитки, употреблять наркотические средства, психотропные или токсические вещества на рабочем месте или в рабочее время; устанавливать системный блок в закрытых объемах мебели, непосредственно на полу; использовать для подключения ПК розетки, удлинители, не оснащенные заземляющим контактом (шиной).

Работники, не выполняющие эти требования, привлекаются к ответственности согласно законодательству.

3.2 Производственная санитария

3.2.1 Микроклимат и его параметры

Параметры микроклимата могут меняться в широких пределах, в то время как необ¬ходимым условием жизнедеятельности человека является поддержание постоянства температуры тела благодаря терморегуляции, т.е. способности организма регу¬лиро¬вать отдачу тепла в окружающую среду. Принцип нормирования микро¬кли¬мата – соз¬дание оптимальных условий для теплообмена тела человека с окружающей средой.

Вычислительная техника является источником существенных тепловыделений, что может привести к повышению температу¬ры и снижению относительной влажности в по¬мещении. В по¬мещениях, где установлены компьютеры, должны соблюдаться оп¬реде¬ленные параметры микроклимата. В санитарных нормах СН–245–71 установлены вели¬чины параметров микроклимата, создающие комфортные условия. Эти нормы ус¬танав¬ливаются в зависимости от времени года, характера трудового процесса и харак¬тера производственного помещения (см. табл. 4.1).

Объем помещений, в которых размещены работники вычис-лительных центров, не должен быть меньше 19,5м3/человека с учетом максимального числа одновременно ра¬ботающих в сме¬ну. Нормы подачи свежего воздуха в помещения, где располо¬жены ком¬пьютеры, приведены в табл. 4.2.

Таблица – 3.1 Параметры микроклимата для помещений, где установлены компьютеры

Период года Параметр микроклимата Величина

Холодный Температура воздуха в помещении 23–25°С

Относительная влажность 45–55%

Скорость движения воздуха До 0.1 м/с

Теплый Температура воздуха в помещении 25–27°С

Относительная влажность 45–65%

Скорость движения воздуха 0.1–0.2 м/с

Таблица – 3.2 Нормы подачи свежего воздуха в помещения, где расположены компьютеры

Характеристика помещения Объемный расход подаваемого в помещение свежего воздуха, м3 / на одного человека в час

Объем до 20 м3 на человека Не менее 35

20–40 м3 на человека Не менее 25

Более 40 м3 на человека Естественная вентиляция

Помещение без окон и световых фонарей Не менее 65

Для обеспечения комфортных условий используются как организационные методы (рациональная организация проведения работ в зависимости от времени года и суток, чередование труда и отдыха), так и технические средства (вентиляция, кондициониро¬вание воздуха, отопительная система).

3.2.2 Освещение

Восприятие света является важнейшим элементом нашей способности действовать, поскольку позволяет оценивать местонахождение, форму и цвет окружающих нас предметов.

Для успешной зрительной работы в условиях изменчивости освещенности большое значение имеет так называемая зрительная адаптация, т.е. приспособление глаза к условиям освещения. Благодаря процессу адаптации зрительный анализатор обладает способностью работать в широком диапазоне освещенностей. Различают световую и темновую адаптации.

Световая адаптация – приспособление глаза к работе в условиях высокой яркости поля зрения.

Световая адаптация при повышении яркостей в поле зрения происходит быстро – в течение 5–10 мин.

Темновая адаптация – приспособление глаза к более низким яркостям поля зрения. Темновая адаптация развивается медленнее – от 30 мин до 2 ч. Процесс адаптации сопровождается фотохимическими и нервными процессами, перестройкой рецептивных полей в сетчатке глаза, изменением диаметра зрачка (зрачковый рефлекс). Частые изменения уровней яркости приводят к снижению зрительных функций, развитию утомления вследствие переадаптации глаза. Зрительное утомление, связанное с напряженной работой и частой переадаптацией, приводит к снижению зрительной и общей работоспособности.

Каждый вид деятельности, связанный с необходимостью различения того или иного объекта, требует определенного уровня освещенности на том участке, где эта деятельность осуществляется. Обычно чем сильнее затруднено зрительное восприятие, тем выше должен быть средний уровень освещенности.

При организации рационального освещения, выборе источников света и светильников учитываются назначение помещения, его размеры и ка¬тегория по взрывопожароопасности, возможные загрязнения (пыль, газы, пары), характеристика и разряд выполняемой работы, нормированная освещенность и цветовая отделка.

Для создания нормальной световой среды применяют различные системы освещения.

Виды освещения производственных помещений и рабочих мест

Освещение как свет от какого–либо источника, создающее осве-щенность поверхностей предметов и обеспечивающее зрительное восприятие этих предметов, бывает:

естественное;

искусственное;

совмещенное.

В системах производственного освещения применяют люминесцентные газоразрядные лампы, имеющие форму цилиндрической стеклянной трубки. Внутренняя поверхность трубки покрыта тонким слоем люминофора, который преобразует ультрафиолетовое излучение газового электрического разряда в видимый свет. Люминесцентные газоразрядные лампы в зависимости от применяемого в них люминофора создают различный спектральный состав света.

Кроме люминесцентных газоразрядных ламп (низкого давления), в производственном освещении применяют газоразрядные лампы высокого давления:

лампы ДРЛ (дуговые ртутные люминесцентные);

галогенные лампы ДРИ (дуговые ртутные с йодидами);

ксеноновые лампы ЛКсТ (дуговые ксеноновые трубчатые), которые в основном применяются для освещения территорий предприятия;

натриевые лампы ДНаТ (дуговые натриевые трубчатые), используемые для освещения цехов с большой высотой (в частности, многих литейных цехов).

Применяются для освещения производственных помещений также лампы накаливания, в которых свечение возникает путем нагревания нити накала до высоких температур. Они просты и надежны в эксплуатации. Недостатками их являются низкая световая отдача (не более 20　лм/Вт), ограниченный срок службы (до 1000 ч), преобладание излучения в желто–красной части спектра, что искажает цветовое восприятие.

В осветительных системах используют лампы накаливания различных типов:

вакуумные (НВ);

газонаполненные биспиральные (НБ);

биспиральные с криптоноксеноновым наполнением (НБК);

зеркальные с диффузно отражающим слоем и др.

Все большее распространение получают лампы накаливания с йодным циклом – галоидные лампы, которые имеют лучший спектральный состав света и хорошие экономические характеристики.

Качественные показатели освещения в производственных помещениях во многом определяются правильным выбором светильников, представляющих собой совокупность источника света и осветительной арматуры.

Для того чтобы обеспечить условия, необходимые для зрительного комфорта, в системе освещения должны быть реализованы следующие предварительные требования: однородное освещение, оптимальная яркость, отсутствие бликов, соответствующая контрастность, правильная цветовая гамма, отсутствие стробоскопического эффекта или мерцания света.

Важно рассматривать свет на рабочем месте, руководствуясь не только количественными, но и качественными критериями — это точность, с которой должны выполняться работы, объем работы, степень перемещений рабочего при работе и т. д. Свет должен включать компоненты как рассеянного, так и прямого излучения. Результатом этой комбинации должно стать тенеобразование большей или меньшей интенсивности, которое должно позволить рабочему правильно воспринимать форму и положение предметов на рабочем месте.

Основной задачей производственного освещения является поддержание на рабочем месте освещенности, соответствующей характеру зрительной работы. Увеличение освещенности рабочей поверхности улучшает видимость объектов за счет повышения их яркости, увеличивает скорость различения деталей, что сказывается на росте производительности труда.

3.2.2.1 Расчет освещенности рабочего места

Расчет освещенности рабочего места сводится к выбору системы освещения, опре¬де¬лению необходимого числа светильников, их типа и размещения. Исходя из этого, рас¬считаем параметры искусственного освещения.

Обычно искусственное освещение выполняется посредством электрических источ¬ни¬ков света двух видов: ламп накаливания и люминесцентных ламп. Будем использо¬вать люминесцентные лампы, которые по сравнению с лампами накаливания имеют ряд су¬щественных преимуществ:

по спектральному составу света они близки к дневному, естественному свету;

обладают более высоким КПД;

обладают повышенной светоотдачей;

более длительный срок службы.

Расчет освещения производится для комнаты площадью 18м2, ширина которой 3м, высота – 3 м. Воспользуемся методом светового потока.

Для определения количества светильников определим световой поток, падающий на поверхность по формуле:

F=(E∙K∙S∙Z)/n (3.1)

где F – рассчитываемый световой поток, Лм;

Е – нормированная минимальная освещенность, Лк (определяется по таблице). Работу специалиста в сфере программирования, в соответствии с этой таблицей, можно отнести к разряду точных работ, следовательно, минимальная освещенность будет Е = 300Лк;

S – площадь освещаемого помещения (в нашем случае S = 13м2);

Z – отношение средней освещенности к минимальной (обычно принимается равным 1,1…1,2 , пусть Z = 1,1);

К – коэффициент запаса, учитывающий уменьшение светового потока лампы в резуль¬тате загрязнения светильников в процессе эксплуатации (его значение зависит от типа помещения и характера проводимых в нем работ и в нашем случае К = 1,5);

n – коэффициент использования, (выражается отношением светового потока, падаю¬щего на расчетную поверхность, к суммарному потоку всех ламп и исчисляется в долях единицы; зависит от характеристик светильника, размеров помещения, окраски стен и потолка, характеризуемых коэффициентами отражения от стен (РС) и потолка (РП)), значение коэффициентов РС и РП были указаны выше: РС = 50%, РП = 70%. Значение n определим по таблице коэффициентов использования различ-ных светильников. Для этого вычислим индекс помещения по формуле:

I=S/(h∙(A+B)) ; (3.2)

где S – площадь помещения, S = 13 м2; h – расчетная высота подвеса, h = 2,5 м;

A – ширина помещения, А = 3 м;

В – длина помещения, В = 5 м.

Подставив значения получим:

I=13/(2,5∙(3+5))=0,65 (3.3)

Зная индекс помещения I, по таблице коэффициент использования светового потока находим n = 0,3

Подставим все значения в формулу для определения светового потока F:

F=(300∙1,5∙13∙1,1)/0,3=21450 Лм (3.4)

Для освещения выбираем люминесцентные лампы типа ЛБ40–1, световой поток кото¬рых F = 4320 Лк.

Рассчитаем необходимое количество ламп по формуле:

N=F/F\_л ; (3.5)

N – определяемое число ламп;

F – световой поток, F = 21450 Лм;

Fл – световой поток лампы, Fл = 4320 Лм.

N=21450/4320=5 шт. (3.6)

При выборе осветительных приборов используем светильники типа ОД. Каждый светильник комплектуется двумя лампами.

3.3 Техника безопасности

3.3.1 Требования безопасности во время эксплуатации ЭВМ и работы с ним

Перед началом работы с ПК работник обязан:

проветрить рабочее помещение;

проверить:

устойчивость положения оборудования на рабочем столе;

отсутствие видимых повреждений оборудования, дискет в дисководе системного блока;

исправность и целостность питающих и соединительных кабелей, разъемов и штепсельных соединений, защитного заземления (зануления);

исправность мебели;

отрегулировать:

положение стола, стула (кресла), подставки для ног, клавиатуры, экрана монитора;

освещенность на рабочем месте. При необходимости включить местное освещение;

протереть поверхность экрана монитора, защитного фильтра (при его наличии) сухой мягкой тканевой салфеткой;

убедиться в отсутствии отражений на экране монитора, встречного светового потока;

включить оборудование ПК в электрическую сеть, соблюдая следующую последовательность: стабилизатор напряжения (если он используется), блок бесперебойного питания, периферийные устройства (принтер, монитор, сканер и другие устройства), системный блок.

Запрещается приступать к работе при:

выраженном дрожании изображения на мониторе;

обнаружении неисправности оборудования;

наличии поврежденных кабелей или проводов, разъемов, штепсельных соединений;

отсутствии или неисправности защитного заземления (зануления) оборудования.

Во время работы с ПК работник обязан:

соблюдать требования охраны труда;

содержать в порядке и чистоте свое рабочее место;

держать открытыми вентиляционные отверстия оборудования;

соблюдать оптимальное расстояние от экрана монитора до глаз.

Работу за экраном монитора следует периодически прерывать на регламентированные перерывы, которые устанавливаются для обеспечения работоспособности и сохранения здоровья, или заменять другой работой с целью сокращения рабочей нагрузки у экрана.

Оператору и пользователю ПК во время работы запрещается:

касаться одновременно экрана монитора и корпуса компьютера или клавиатуры;

прикасаться к задней панели системного блока компьютера при включенном питании;

переключать разъемы интерфейсных кабелей периферийных устройств;

при включенном питании:

выполнять перезагрузку компьютера;

самостоятельно вскрывать и производить ремонт компьютеров;

класть посторонние предметы на клавиатуру, монитор, системный блок, мышь и другое компьютерное оборудование, а также на электрощиты, электропроводку и другое электрическое оборудование.

Время регламентированных перерывов в течение рабочего дня (смены) устанавливается в зависимости от его (ее) продолжительности, вида и категории трудовой деятельности.

Продолжительность непрерывной работы с ПК без регламентированного перерыва не должна превышать 2 часов.

Во время регламентированных перерывов для снижения нервно–эмоционального напряжения, утомления зрительного анализатора, улучшения функционального состояния нервной, сердечнососудистой, дыхательной систем, а также мышц плечевого пояса, рук, спины, шеи и ног целесообразно выполнять комплексы упражнений.

Работникам с высоким уровнем напряженности труда во время регламентированных перерывов и в конце рабочего дня показана психологическая разгрузка в специально оборудованных комнатах психологической разгрузки.

С целью уменьшения отрицательного влияния монотонности необходимо применять чередование операций.

При работе с текстовой информацией следует отдавать предпочтение физиологически наиболее оптимальному режиму представления черных символов на белом фоне.

3.3.2 Требования безопасности после окончания работы

После окончания работ специалист по защите информации и пользователь ПК обязан:

корректно закрыть все активные задачи;

при наличии дискеты в дисководе извлечь ее;

выключить питание системного блока;

выключить питание всех периферийных устройств;

отключить блок бесперебойного питания;

отключить питающий кабель от сети;

осмотреть и привести в порядок рабочее место;

о неисправностях оборудования и других замечаниях по работе с ПК сообщить непосредственному руководителю или лицам, осуществляющим техническое обслуживание оборудования;

при необходимости вымыть с мылом руки.

В данном разделе были рассмотрены основные положения и нормы охраны труда, необходимые для работы техника по защите информации.

3.3.3 Электробезопасность

Электричество опасно не только своим поражающим воздействием, но и фактором нарушения производственного процесса, выведением оборудования из строя и опасность возгорания. Обеспечение безопасного режима работы в таких установках является неотъемлемой и важной составляющей работы специалиста по охране труда и всего руководства организации в целом. Законодательство Республики Казахстан жестко регламентирует порядок проведения таких работ, сроки реализации обучающих мероприятий, категории и разряд для получения доступа к установкам повышенной опасности и иные аспекты. Поэтому соблюдение таких правил обеспечивает не только соответствие нормам закона, но и значительно уменьшает вероятность негативных последствий.

В целом охрана труда электробезопасность представляет собой комплекс мер, которые на постоянной основе должны осуществляться на предприятии и подвергаются жесткому надзору и контролю со стороны государственных органов. Эти мероприятия позволяют обеспечить режим безопасности и непрерывности производства. Также нормативным закреплением данного режима на объекте занимаются специальные службы, ответственные за соблюдение норм по охране труда.

Основные правила электробезопасности:

 запрещается использовать неисправные электроприборы;

 запрещается использовать неисправные розетки и вилки;

 запрещается подходить в оголенным проводам ближе, чем на 6–8 метров;

 запрещается браться за провода или корпуса работающего оборудования мокрыми руками;

 запрещается загромождать электрощит, электрощитовые проходы.

Также в случае обнаружения во время проведения проверки государственными надзорными органами несоответствия требованиям законодательства, различных нарушений и нештатных ситуаций ответственные лица несут административную, уголовную и гражданско–правовую ответственность за такие нарушения.

Причины поражения электрическим током:

 прикосновение к токоведущим частям электроустановок, проводам, находящимся под напряжением;

 появление напряжение на металлических нетоковедущих частях установок, корпусах, кожухах;

 появление напряжения на отключенных токоведущих частях во время ремонтных работ и осмотров (кто–то включил рубильник и подал напряжение в сеть);

 возникновение шагового напряжения на участке земли, где находится человек;

 возникновение электрической дуги в зоне работы человека (короткое замыкание).

Основные меры защиты от поражения электрическим током:

 защитное заземление;

 зануление;

 обеспечение недоступности токоведущих частей для случайного прикосновения;

 применение малого напряжения;

 защитное отключение.

Общие требования электробезопасности:

 эксплуатацию электроустановок должен осуществлять подготовленный электротехнический персонал;

 все виды электроустановок должны быть закрыты на ключ;

 электротехнический и не электротехнический персонал должны проходить соответствующие инструктажи по электробезопасности;

 в помещениях, в которых используется напряжение двух и более. номиналов, на всех штепсельных розетках должны быть надписи с указанием номинального напряжения. Использование автотрансформаторов для питания светильников сети 12–50 В не разрешается. Применение для переносного освещения люминесцентных ламп, не укрепленных на жестких опорах, не допускается;

 токоведущие провода (сварочные провода, провода удлинители переносных ламп) прокладываются так чтобы их не могли повредить машины и механизмы.

Выполнение работ со снятием напряжения:

 отключить электроустановку

 повесить запрещающие плакаты

 проверить отсутствие напряжения

 установить заземление

 оградить рабочее место, вывесив плакаты безопасности

 по окончанию работ привести рабочее место в порядок, снять временные ограждения и переносные плакаты безопасности, заземления, закрыть дверь электроустановки на замок.

3.4 Пожарная безопасность

Охрана труда и пожарная безопасность промышленных предприятий – важный комплекс мероприятий, обеспечивающий сохранение здоровья работников промышленности. Подобные правила разрабатываются и утверждаются специальными комиссиями, деятельность которых направлена на предотвращение несчастных случаев на рабочих местах.

Чтобы обеспечить безопасность труда и пожарную безопасность, необходимо решить следующие задачи:

 провести разработку и осуществить внедрение этой системы. Руководство фирмы и ответственные лица должны разработать и провести внедрение правил, соблюдение которых обязательно для сотрудников предприятия. Все работники обязаны пройти соответствующее обучение;

 обеспечить контроль за аварийностью цехов и оборудования. Необходимо периодическое проведение проверок и принятие соответствующих мер;

 обеспечение противопожарной безопасности при непосредственной работе с техникой, эксплуатацией помещений в слесарных и других цехах;

 оснащение подразделений фирмы техникой пожаротушения. Проверка исправности и поддержание постоянной готовности к применению в случае необходимости. Предоставление транспорта в случае необходимости. Наличие исправных систем противопожарной сигнализации и обеспечение их работоспособности;

 важно обеспечить план по финансовому обеспечению принимаемых мер защиты;

 проведение обучения сотрудников правильным действиям в случае возникновения пожара, проведение инструктажей, организация стендов с образовательными материалами и тематических лекций;

 проведение регулярных проверок исправности электросети и ее безопасности.

Для организации системы противопожарной безопасности необходимо обеспечить соответствующий документооборот:

 на предприятии должны быть оформлены приказы о назначении ответственных лиц в этой сфере;

 необходимо предоставить эвакуационные планы на случай пожара. Они должны быть подготовлены и оформлены для каждого этажа здания. Документы необходимо вывесить на видных местах;

 оформляются объявления о проведении тематических консультаций и графики проверки знаний сотрудников;

 необходимо подготовить документы, связанные с контролем безопасности электрических сетей на предприятии;

 руководство обязано предоставить сотрудникам пособия, программы обучения, тренажеры для отработки навыков в этой сфере;

 средства тушения пожаров, имеющиеся на предприятии, должны быть обеспечены сертификатами, подтверждающими исправность и работоспособность;

 знания сотрудников должны проходить регулярную проверку. Лицам, которые ее прошли, необходимо предоставить подтверждающие документы;

 необходимо обеспечить наличие инструкций о порядке действий при тушении пожара в различных ситуациях.

Проведение инструктажа по пожарной безопасности. Целью инструктажа являются:

 информирование работников о том, каковы главные требования противопожарной безопасности;

 изучение проблем, связанных с технологическими процессами и оборудованием с учетом требований здравоохранения;

 отработка умений правильно пользоваться средствами пожаротушения, имеющимися на предприятии;

 обучение тому, как действовать, если возникнет пожар.

Инструктаж предполагает, что нужно:

 ознакомить с действующими правилами зданий и территории, устройством внутреннего и наружного трубопроводов, работы систем оповещения, проведением эвакуации сотрудников;

 довести до людей актуальные требования, связанные с противопожарной опасностью оборудования, технологических процессов, помещений;

 объяснить, какие меры должны приниматься для обеспечения безопасности в ежедневной деятельности, а также в случае проведения взрывных работ на специальных площадках;

 рассказать об ограничениях, относящихся к применению открытого огня и работам, которые с ним связаны.

Необходимо довести до сотрудников их обязанности в случае возгорания, а также порядок вызова работников пожарной службы.

Соблюдение правил безопасности поможет сохранить здоровье или жизнь сотрудникам предприятия. Если в компании этому вопросу уделяется достаточно внимания, то вероятность пострадать от возгорания практически отсутствует.