**Введение**

В настоящее время в РФ реализуется государственная программ по переходу на цифровую экономику. Эти изменения коснулись и сферу образования. Сегодня образовательные учреждения обязаны иметь сайт образовательной организации, на котором размещена вся актуальная информация о ее деятельности. К тому же данный ресурс позволяет осуществлять имиджевую, рекламную, маркетинговую и образовательную деятельность, а также осуществлять взаимодействие с участниками образовательного процесса.

На сегодняшний день в РФ выпускники общеобразовательных школ имеют право поступать в учреждения среднего профессионального образования и высшего образования. Согласно государственного законодательства данные учреждения могут принимать заявления от абитуриентов в электронном виде. Это факт говорит о том, что существует необходимость создания дистанционной формы подачи документов абитуриентами в образовательные учреждения, так как это упростит работу приемной комиссии, сэкономит время приемной комиссии и абитуриента. Но на сегодняшний день образовательные учреждения таким инструментом не пользуются, в том числе и наше образовательное учреждение.

Объектом исследования является работа приемной комиссии.

Предметисследования является обеспечение дистанционной подачи документов на зачисление абитуриентами в образовательные учреждение.

Целью курсовой работы является разработка сайта для абитуриентов, поступающих в образовательные учреждения.

Исходя из поставленной цели, были поставлены следующие задачи:

* изучить и проанализировать предметную область;
* разработать структуру сайта;
* разработать web-сайт;
* разработать программную документацию.

Практическая значимость курсовой работы заключается в том, что данная разработка сайта позволит абитуриентом подавать заявки на поступление в образование учреждение дистанционно.

**СОДЕРЖАНИЕ**

**Глава 1.** **Теоретическая часть**

1.1 Описание предметной области

**1.1.1 Назначение разработки**

Программное обеспечение предназначено для дистанционного заполнения и заявки на обучение в образовательные учреждения путем заполнения формы на сайте.

**1.1.2. Требования к функциональным характеристикам**

Система должна включать в себя следующие функции:

1. заполнение формы личными данными абитуриента;

* Ф.И.О.;
* пол;
* дата рождения;
* место жительства.

1. Заполнение формы контактными данными абитуриента;

* E-mail адрес;
* контактный телефон;

1. выбор формы обучения;
2. добавление ксерокопии документов (паспорт, аттестат);
3. выбор специальности/специальностей( не более 3-х) на базе 9/11классов;

Входные данные:

* данные пользователя;

Выходные данные:

* e-mail-сообщение формы администратору.

1.1.3. Требования к надежности и безопасности

В программном обеспечении должен быть обеспечен надлежащий уровень защиты информации в соответствии с законом о защите персональной информации от несанкционированного доступа - № 152-ФЗ от 27.07.2006 «О персональных данных». Для работы с приложением необходимо стабильное Интернет-соединение.

1.1.4. Требование к составу и параметрам

Минимальные системные требования для программного обеспечения:

* процессор Intel Pentium 233 МГц или выше;
* стабильное Интернет-соединение.

**1.1.5. Требования к информационной и программной совместимости**

Для работы с программой потребуются:

* операционная система: Windows, Linux, OS X, Android, iOS, Windows Mobile;
* для корректного отображения понадобится любой современный веб-браузер.

ГЛАВА 2. Требования к программе

2.1. Анализ требований и определение спецификаций программного обеспечения

2.1.1. Выбор технологии проектирования

Для эффективного взаимодействия разработчиков и заказчиков применяются различные инструментальные и методологические средства. Одними из самых наглядных средств являются диаграммы. Диаграммы необходимы для того, чтобы показать в более удобной форме, то, какую функцию выполняет программное обеспечение. Наиболее востребованными методологиями, в которых используются метод диаграмм, на данный момент являются SADT и UML.

Существуют стандартизированные, опробованные временем методологи и инструментальные средства, с помощью которых можно исследовать объект и построить его модель.

Основа многих современных методологий моделирования бизнес-процессов составляет методология SADT семейство стандартов IDEF и алгоритмические языки.

Для описания бизнес-моделей чаще всего используется стандарт IDEF0, который предназначен для описания высокоуровневого описания бизнеса компании в функциональном проекте.

Одно из отличий методологии IDEF0 это ее упор на сходство объектов. В IDEF0 разбираются логические отношения между действиями, а не их временная прогрессия.

Главное достоинство анализа бизнес-процессов предприятия, это то, что руководитель или руководство предприятия, которые внедрили у себя конкретную методологию, будут иметь информацию, которая позволит совершенствовать свое предприятие и анализировать будущее.

Язык UML представляет собой язык визуального моделирования, который разработан для спецификации, визуализации, проектирования и документирования компонентов программного обеспечения, бизнес-процессов и других систем. Язык UML является одновременно простым и мощным средством моделирования, который может быть эффективно использован для построения концептуальных, логических и графических моделей сложных систем различного целевого назначения.

Конструктивное использование языка UML основывается на понимании общих принципов моделирования сложных систем и особенностей процесса объектно-ориентированного проектирования (ООП) в частности. Выбор выразительных средств для построения моделей сложных систем предопределяет те задачи, которые могут быть решены с использованием данных моделей. При этом одним из основных принципов построения моделей сложных систем является принцип абстрагирования, который предписывает включать в модель только те аспекты проектируемой системы, которые имеют непосредственное отношение к выполнению системой своих функций или своего целевого предназначения. При этом все второстепенные детали опускаются, чтобы чрезмерно не усложнять процесс анализа и исследования полученной модели.

Другим принципом построения моделей сложных систем является принцип многомодельности. Этот принцип представляет собой утверждение о том, что никакая единственная модель не может с достаточной степенью адекватности описывать различные аспекты сложной системы. Применительно к методологии ООП это означает, что достаточно полная модель сложной системы допускает некоторое число взаимосвязанных представлений, каждое из которых адекватно отражает некоторый аспект поведения или структуры системы. При этом, наиболее общими представлениями сложной системы принято считать статическое и динамическое представления, которые в свою очередь могут подразделяться на другие более частные представления.) феномен сложной системы как раз и состоит в том, что никакое ее единственное представление не является достаточным для адекватного выражения всех особенностей моделируемой системы.

Еще одним принципом прикладного системного анализа является принцип иерархического построения моделей сложных систем. Этот принцип предписывает рассматривать процесс построения модели на разных уровнях абстрагирования или детализации в рамках фиксированных представлений. При этом исходная или первоначальная модель сложной системы имеет наиболее общее представление (метапредставление). Такая модель строится на начальном этапе проектирования и может не содержать многих деталей и аспектов моделируемой системы, также в UML используются следующие диаграммы для описания программного обеспечения:

* диаграмма прецедента;
* диаграмма классов;
* диаграмма действий для программы;
* диаграмма действий для пользователя;
* диаграмма развертывания;
* диаграмма компонент;
* размещенная диаграмма компонента на диаграмме развертывания.

Для разработки программного обеспечения «Форма дистанционной подачи заявки на поступление» была выбрана методология UML. Данная модель объектно-ориентирована, в результате чего методы описания результатов анализа и проектирования близки к методам программирования на современных объектно-ориентированных языках, а IDEF0 больше направлена на организацию бизнес-процессов.

2.1.2. Построение моделей

Диаграмма прецедентов

Пользователь заполняет форму своими данными, выбирает форму обучение на которой хочет обучаться(очная/заочная), пишет свой контактный телефон и адрес электронной почту, прикрепляет сканкопию документов(паспорта, аттестата), выбирает одну или более специальностей на которые хочет поступить и отправляет на e-mailучреждения.(рис. 1)

Сервер

Пользователь

Приемная комиссия

Рисунок 1 – Диаграмма прецедентов

Диаграмма классов

Пользователь заполняет заявку на поступление, отправляет администратору, после чего администратор рассматривает заявку и связывается с пользователем. (рис. 2)

Рисунок 2 – Диаграмма классов

1

1

1

1

1

1

**1**

**Пользователь**

Контактные данные

Ввод

**Web сайт**

Информация

Предоставление информации и отправка данных администратору

Обработка данных

**Администратор**

**1**

Заполнение формы

Отправка данных

**1**

Приемная комиссия

Проверка заявок

Диаграмма действий для программы

На данной модели показан принцип работы приложения. (рис. 3)

Рисунок 3 – Диаграмма действий для программы

Заполнение формы пользователем

Форма заполнена корректно?

Отправка на сервер

Проверка приемной комиссией

нет

да

Рисунок 3 – Диаграмма действий для программы

Диаграмма действий для пользователя

На данной модели показан принцип работы пользователя с программой. (рис.4)

Заполнение формы абитуриентом

Отправка заявление на почту учреждения

Ожидание ответа

Рисунок 4 – Диаграмма действий для пользователя

Диаграмма развертывания

Данная модель показывает, какие устройства используются в данном программном обеспечении.(рис.5)

Клиентский компьютер

Сервер с приложением

Интернет

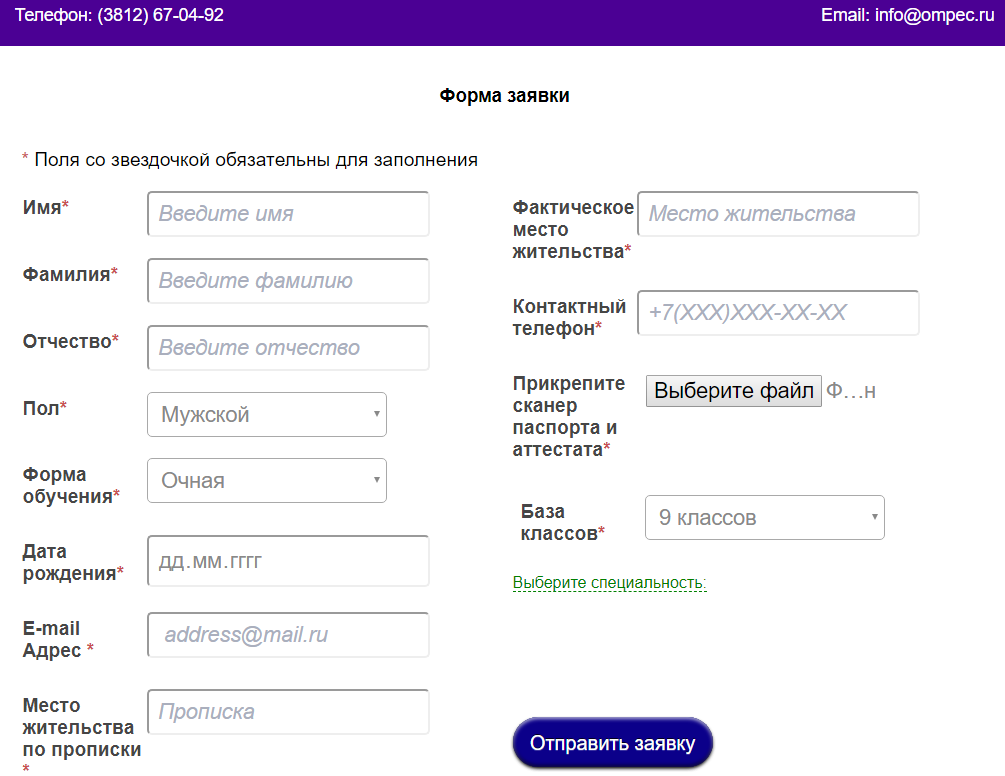
Приемная комиссия

Рисунок 5 – Диаграмма развертывания

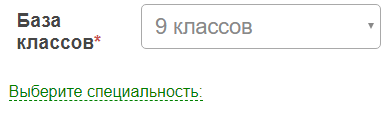
2.2. Детальное проектирование программного обеспечения

Разрабатываемая программа предположительно будет состоять из одной главной страницы с формой заявки.

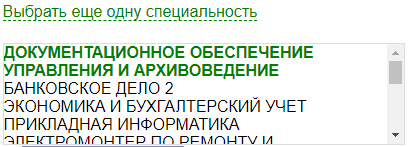
Форма заявки – главная страница приложения, реализована в файле index.html. В ней пользователь заполняет свои данные



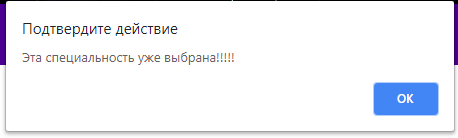
Отображение списка специальностей реализовано на JQuery. JQuery –библиотека JavaScript фокусирующая на взаимодействии JavaScript и html.



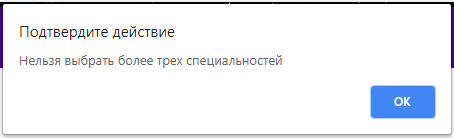
При выборе базы классов абитуриента, ему будут предложены определенные специальности.



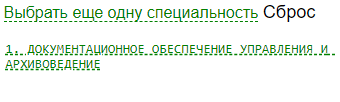
Абитуриент может выбрать одну или более специальностей (но не более 3-х) для подачи заявки. Если абитуриент по ошибки выберет специальность, которую уже выбрал, то программа выведет окно ошибки.



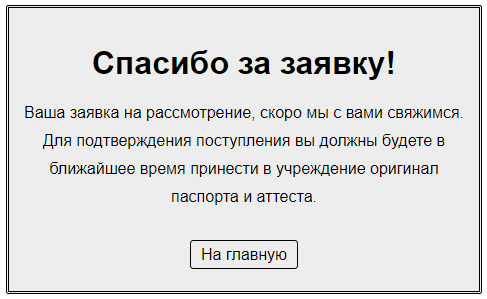
Если абитуриент выбрал больше трех специальностей то программа выведет ошибку:



При выборе не тех специальностей у пользователя есть возможность сбросить выбранные специальности:

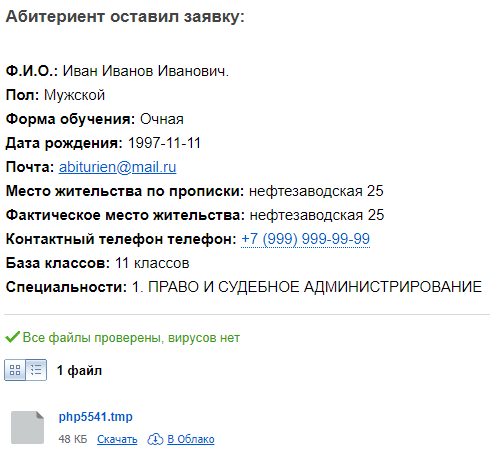


После отправки формы, пользователь переходит на страницу благодарности:



По нажатию на кнопку «На главную» пользователь переходит обратно к форме.

Администратор получает письмо с заявкой абитуриента:



ГЛАВА 3. реализация

3.1. Обоснование выбора средств разработки

Для реализации данного программного продукта был выбран принцип веб-приложения.

Веб-приложение – клиент-серверное приложение, в котором клиентом выступает браузер, а сервером – веб-сервер. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются кроссплатформенными сервисами.

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

Для написания и редактирования программного кода использовался кроссплатформенный текстовый редактор Sublime Text 3.

Таким образом, с помощью выбранных средств разработки, необходимо реализовать программу, а также следует описать программный продукт.

3.2 Описание основных программных модулей

При разработке данного приложения использовались такие языки программирования и разметки, как:

JavaScript – объектно-ориентированный сценарный язык программирования. Является реализацией языка ECMAScript, обычно используется как встраиваемый язык для программного доступа к объектам приложений;

PHP – скриптовый язык общего назначения, интенсивно применяемы в разработке веб-приложений. Является одним из самых популярных языков программирования;

HTML – язык гипертекстовой разметки, стандартизированный язык разметки документов во Всемирной паутине;

CSS – каскадная таблица стилей, формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

Заключение

Результатом разработки дипломного проекта является программа «Форма дистанционной подачи заявки на обучение».

В процессе выполнения дипломной работы была определена структура проекта, которая включает: назначение, совместимость, функции, надежность и безопасность, а также технические средства разрабатываемой системы. Были проанализированы требования к системе и определены спецификации ПО, спроектирована модель данных, и описан интерфейс программного продукта.

После всестороннего анализа системы была реализована система с помощью выбранных средств разработки и описаны ее основные программные модули.

Таким образом, поставленные задачи были выполнены, и разработанное приложение предоставляет пользователям, возможность подавать заявки на обучение дистанционно.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. [Марин Хауэрбеке: Элегантный JavaScript](http://eloquentjavascript.net/),2014. – 464 с.
2. программированию. Обработка числовых данных, 2008. – 288 с.
3. Линн Бейли.: Изучаем PHP и MySQL, 2010. – 54 с.
4. Steave Suehring, Janet Valade.: PHP, MySQL, JavaScript & HTML5 All-in-One For Dummies, 2013. – 129 с.
5. [Джон Резиг](https://www.google.ru/search?newwindow=1&sa=X&biw=1680&bih=920&q=%D0%94%D0%B6%D0%BE%D0%BD+%D0%A0%D0%B5%D0%B7%D0%B8%D0%B3&stick=H4sIAAAAAAAAAOPgE-LRT9c3NErKM7K0NDdQ4tLP1TcwLrQsMsjSkslOttJPys_P1i8vyiwpSc2LL88vyrZKLC3JyC8CAOs3Z4g6AAAA&ved=0ahUKEwjkoa_RoMrUAhUkApoKHYhaBncQmxMIpAIoATAO), [Бэар Биболт](https://www.google.ru/search?newwindow=1&sa=X&biw=1680&bih=920&q=%D0%91%D1%8D%D0%B0%D1%80+%D0%91%D0%B8%D0%B1%D0%BE%D0%BB%D1%82&stick=H4sIAAAAAAAAAOPgE-LRT9c3NErKM7K0NDdQ4tLP1TcwqSw0Mq7SkslOttJPys_P1i8vyiwpSc2LL88vyrZKLC3JyC8CAB_bhoE6AAAA&ved=0ahUKEwjkoa_RoMrUAhUkApoKHYhaBncQmxMIpQIoAjAO).: Секреты JavaScript ниндзя, 2013. – 298 с.
6. Осипов Д.Л.: Программирование для Windows, OS X, iOS, 2014. – 244 с.
7. Дэвид Фланаган.: JavaScript. Подробное руководство, 2013. - 122
8. Юрий Шпак.: Web-разработка средствами ModX, 2012. – 44
9. Джим Моррисон.: Изучаем HTML и CSS, 2009. – 228
10. БХВ-Петербург, 2010. – 298 с.
11. Справочник html/css [Электронный ресурс] URL: http://htmlbook.ru/.