## 2.2. Техническое задание

### 2.2.1. Общие сведения

В данной главе представлены общие сведения о системе:

1. полное наименование системы и ее условное обозначение: web-сайт территориально-производственного предприятия «Урайнефтегаз»
2. шифр темы или шифр (номер) договора: данный программный продукт разрабатывался вне рамок какого-либо договора, а в рамках дипломного проектирования.
3. наименование предприятий разработчика и заказчика системы, их реквизиты:
4. перечень документов, на основании которых создается информационная система: нет
5. плановые сроки начала и окончания работ: 1 апреля 2010 года — 30 мая 2010 года;
6. сведения об источниках и порядке финансирования работ: данные работы проводятся на безвозмездной основе;
7. порядок оформления и предъявления заказчику результатов работ по созданию системы, ее частей и отдельных средств: по окончанию работ заказчику должны быть предъявлены и переданы диск с данными проекта, набор инструкций и документации для полноценного функционирования сайта.

### 2.2.2. Назначение и цели создания системы

Видом автоматизируемой деятельности является распространение информации, получение обратной связи. Данную систему предполагается использовать в организации.

Целью создания системы является всестороннее представление предприятия, как на местном, так и на федеральном уровне. В следствии - привлечение молодых квалифицированных специалистов, организация обратной связи с сотрудниками предприятия, взаимодействие со СМИ.

### 2.2.3. Характеристика объектов автоматизации

Объектом автоматизации являются:

* получение пользователями внутренних документов предприятия,
* размещение контактной информации,
* размещение информации о вакансиях,
* размещение данных о производственной деятельности предприятия,
* размещение внутренних нормативных документов.

Требований к условиям эксплуатации и характеристиках окружающей среды нет.

### 2.2.4. Требования к системе в целом

***а) Требования к структуре и функционированию системы.***

Разработанная система сайта должна состоять из следующих подсистем:

* доступа к БД MySQL
* пользовательского интерфейса
* визуализации данных
* хранения данных
* классов уровня приложения

Подсистема доступа к БД должна позволять легко изменять структуру БД без изменения большей части программы, обеспечить корректную загрузку и выгрузку данных, правильную работу запросов и снижать вероятность ошибок.

Подсистема пользовательского интерфейса должна обеспечивать динамическую сборку страниц, легкость корректировки отдельных частей и отсутствие конфликтов между ними, правильный вывод страниц.

Подсистема визуализации данных должна корректно и визуально привлекательно для пользователя предоставлять информацию.

Структура таблиц БД, каталогов и файлов подсистемы хранения данных должна быть разработана таким образом, что бы легко было найти и изменить хранящуюся в них информацию.

Подсистема классов должна строится таким образом, чтобы каждый класс обладал хорошо определённой сферой ответственности. Код должен быть снабжен комментариями и был понятен другому программисту.

В проекте должно быть несколько режимов функционирования. Главные – это режим администратора и режим пользователя.

В перспективе развитие системы должно быть легко осуществимо. Возможно добавление новых сервисов.

***б) Требования к персоналу.***

Для функционирования системы необходим 1 человека: администратор.

Системный администратор должен уметь работать с оболочкой phpMyAdmin, обладать знаниями по web-программированию, в частности на языке php. А также выполнять роль модератора при проверке комментариев и добавлять в базу новые данные.

Пользователь должен обладать знаниями об использовании браузера.

***в) Требования к системе.***

Система должна быть надежной: отказоустойчивой к ошибкам пользователя (нарушения правильной последовательности действий), ошибкам в загружаемых данных. Также система должна восстанавливаться после различных сбоев (отключение электричества, сбой в работе операционной системы, оборудования т т.д.). За данный функционал отвечает сервер MySQL

Программный комплекс должен быть кросс-платформенен, легко мигрировать на разные архитектуры серверов и операционные системы.

Продукт должен обладать эргономичным и интуитивно-понятным интерфейсом.

Должна быть предусмотрена возможность обновления и обслуживания системы, в том числе и сторонними разработчиками. Возможность расширения функционала приложений.

Должна присутствовать аутентификация пользователя на сервере, для разграничения доступа к информации и ее защите от изменений, настройки доступа к БД.

### 2.2.5. Требования к функциям (по подсистемам)

Перечень подлежащих автоматизации задач:

* Поступление сообщений с вопросами от пользователей,
* Организация обратной связи,
* Возможность скачивания внутренних документов предприятия.

### 2.2.6. Перечень критериев и отказов

*Фатальные ошибки.* Сбои в системе, которые приводят к потере ее работоспособности или к порче данных. Ошибка управления или попытка взлома системы. Ошибка авторизации.

*Серьезные ошибки.* Ошибки, которые не позволяют достичь желаемого результата при использовании некоторого средства или функции программного продукта. Ошибка выполнения SQL-запроса. Ошибка запроса открытия файла.

*Маленькие ошибки.* Ошибки, которые вызывают неудобство работы с системой, например, ошибки в задании порядка перехода от одного элемента управления к другому в форме ввода информации.

*Незначительные ошибки.* К данному типу ошибок относятся орфографические ошибки в надписях или не выровненные элементы управления. Ошибки ввода данных в форму.

### 2.2.7. Требования к видам обеспечения.

Система предъявляет следующие требования к видам обеспечения:

а) **информационному:** для хранения данных использовать СУБД MySQL, запущенную на удаленном, либо этом же компьютере. Для выполнения скриптов, написанных на языке PHP понадобится интерпретатор PHP. Для управления базами данных phpMyAdmin.

в) **лингвистическому:** использовать язык программирования PHP, язык запросов SQL, язык описания HTML, язык разметки CSS, XML.

г) **программному и техническому:**

Характеристики компьютера пользователя:

* операционная система: Windows/Linux/MacOS/Solaris;
* Intel Pentium III/Celeron 800  МГц ;
* оперативная память 256 Мбайт;

Сервер баз данных:

* операционная система: Windows/Linux/MacOS/Solaris;
* Intel Pentium III/Celeron 800  МГц;
* оперативная память 256 Мбайт;

Необходимый набор программ пользователя:

* браузер Internet Explorer/ Opera и др.;

Необходимый набор программ администратора:

* браузер Internet Explorer/ Opera и др.;
* phpMyAdmin;
* текстовый редактор

е) **организационному:**  для функционирования системы необходим 1 человека: администратор.

Системный администратор должен уметь работать с оболочкой phpMyAdmin, обладать знаниями по web-программированию, в частности на языке php. А также выполнять роль модератора при проверке комментариев и добавлять в базу новые данные.

ж) **методическому (состав нормативно-технической документации):**

Инструкция программисту

Инструкция администратору

Инструкция пользователю

### 2.2.8. Перечень этапов работ и сроки исполнения

Таблица 1 Этапы и сроки выполнения проекта

| **Период учебного процесса** | **Кол-во недель** | **Ориентировочные сроки** | **Выполняемые работы** |
| --- | --- | --- | --- |
| дипломное  проектирование | 2 недели | Апрель-Май | систематизация материала; описание бизнес-процессов, формулировка постановки задачи, подготовка обзора состояния вопроса |
|  | 2 недели | решение задач экономического раздела дипломного проекта |
|  | 6 недель | проектирование и разработка предметно-ориентированной информационной системы |
|  | 2 недели | оформление проекта |
| Защита | 2 недели | Июнь | защита проекта перед Государственной  аттестационной комиссией |

### 2.2.9. Состав исполнителей работ

Работы по созданию проекта проводились физическим лицом - студентом УлГТУ, Кандрушиным Алексеем Олеговичем. Консультантом по разработке являлся к.э.н. доцент кафедры «Информационные системы», руководитель дипломного проекта Суркова Елена Викторовна.

### 2.2.10. Порядок контроля системы

Для контроля необходимо провести следующие виды тестирования:

*Функциональное тестирование.* Оно включает проверку каждой команды меню, панели инструментов и каждой операции, которую выполняет система.

*Тестирование приложения.* При выполнении тестирования приложения испытатель моделирует действия пользователя. Моделируя различные комбинации действий, можно найти ошибки, относящиеся к интеграции компонентов, а также недоработки, которые не были обнаружены при функциональном тестировании.

*Тестирование на скорость выполнения.* Измеряется время отклика системы на некоторое действие, документа. Обычно при этом определяются параметры эффективности. Одновременно с измерением времени на выполнение запроса необходимо записать также размер базы данных.

*Предельное тестирование.* Позволяет определить пропускную способность программного продукта. Запускаются задачи, увеличивая размеры файлов или баз данных, и таким образом выявляют предельное значение объема данных, при котором завершение выполнения задачи невозможно. Максимальный объем данных является существенным ограничением при программировании.

### 2.2.11. Перечень подлежащих разработке документов

* Инструкция пользователю
* Инструкция программисту
* Инструкция системному администратору