

# **СОДЕРЖАНИЕ**

## ВВЕДЕНИЕ 3

## ГЛАВА 1. АНАЛИТИЧЕСКИЙ ОБЗОР ПРОЕКТИРОВАНИЯ ЭКОНОМИЧЕСКИХ СИСТЕМ 5

### 1.1. Анализ предметной области 5

### 1.2. Обзор литературы 6

### 1.3. Обзор существующих решений 8

### 1.4. Описание главных процессов информационной системы 11

### 1.5. Описание требований к информационной системе мониторинга и анализа данных о продажах питьевой воды 14

### 1.6. Определение трудоемкости разработки программы 16

## ГЛАВА 2. ПРОЕКТИРОВАНИЕ ЭКОНОМИЧЕСКОЦ ИНФОРМАЦИОННОЙ СИСТЕМЫ МОНИТОРИНГА И АНАЛИЗА ДАННЫХ О ПРОДАЖАХ ПИТЬЕВОЙ ВОДЫ 18

### 2.1. Диаграмма прецедентов 18

### 2.2. Прецеденты информационной системы 19

### 2.3. Архитектура информационной системы 28

## ГЛАВА 3. РЕАЛИЗАЦИЯ ЭКОНОМИЧЕСКОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ МОНИТОРИНГА И АНАЛИЗА ДАННЫХ О ПРОДАЖАХ ПИТЬЕВОЙ ВОДЫ 34

### 3.1. Разработка информационной системы мониторинга и анализа данных о продажах питьевой воды 34

### 3.2. Интерфейс информационной системы мониторинга и анализа данных о продажах питьевой воды 38

### 3.3. Тестирование информационной системы мониторинга и анализа данных о продажах питьевой воды 44

## ЗАКЛЮЧЕНИЕ 46

## **ВВЕДЕНИЕ**

Люди не могут обходиться без воды ни дня, ежедневно они совершают покупки от маленького до большого объема. В настоящее время, из-за плохого качества воды, люди проводят себе отдельный трубопровод для питьевой воды. Соответственно, как и в любом бизнесе должен присутствовать контроль продукта, что говорит о том, что компании должны анализировать свои продажи и делать выводы. Для того чтобы все подсчеты были верны и совершались за наименьший промежуток времени требуется автоматизированная информационная система, которая будет собирать данные по потреблению воды с предоставлением возможности работы с ними. На сегодняшний день есть большое количество возможных решений, но каждое обладает некоторыми недостатками, которые мы рассмотрим ниже. Актуальность нашей работы заключается в создании автоматизированной информационной системы для мониторинга и анализ данных о продажах воды, в котором будут исправлены все недочеты существующих решений, а также которое будет удобно как для потребителей, так и для владельцев бизнеса по продажам питьевой воды.

Для эффективной работы системы предлагается следующий метод работы - должен будет осуществляться ежедневно сбор данных по потреблению воды с ftp-серверов, с помощью разбора csv файлов и выгрузкой данных в БД. Данные должны быть представлены как в табличном, так и графическом формате. Далее должен быть осуществлен анализ собранной информации, а именно статистика по потреблению за конкретный промежуток времени в табличном и в графическом виде, расчет финансовых показателей, вывод имеющихся тарифов на потребление.

Объект исследования – мониторинг и анализ данных о продажах питьевой воды.

Предмет исследования – процесс автоматизации мониторинга и анализа данных продаж питьевой воды.

Целью работы является проектирование и разработка информационной системы мониторинга и анализа данных о продажах питьевой воды. Для того чтобы достигнуть данной цели, необходимо решить следующие задачи:

1. Проанализировать сферу продаж:

1. Изучить методы анализа продаж, которые подойдут к нашей работе.

2. Изучить актуальные способы мониторинга продаж.
2. Провести обзор существующих решений.
3. Разработать архитектуру информационной системы и алгоритм работы сервера.
4. Выполнить последовательно все этапы разработки программного продукта.

Практическая значимость работы состоит в том, что разработанная информационная система может использоваться для анализа и мониторинга продаж питьевой воды в многоквартирных домах и на предприятиях с любого устройства и локации.

## ГЛАВА 1. АНАЛИТИЧЕСКИЙ ОБЗОР ПРОЕКТИРОВАНИЯ ЭКОНОМИЧЕСКИХ СИСТЕМ

### **Анализ предметной области**

На сегодняшний день не существует проблемы в получении питьевой воды в России, её можно приобрести в магазинах, в специальных точках разлива, заказать на дом, но самым удобным вариантом является проведение отдельного трубопровода с питьевой водой. Устанавливается специальный кран, расход ресурса контролируется специальным счётчиком.

Все действия по учёту расхода ресурсов управляющий персонал совершает собственноручно. Для начала составляется специальный договор, заносится в базу все данные о потребителе, затем ежемесячно фиксируются показатели потребления, после чего вновь записываются в специальные документы, затем выписывается квитанция, по которой осуществляется позже оплата пользователем.

Автоматизированная система по учёту питьевой воды облегчила бы жизнь, как управляющему персоналу, так и потребителям. Она бы смогла автоматически загружать все данные по расходу в таблицу. Доступ к системе был бы с любой точки при наличии интернета. Система бы предоставляла статистику по потреблению пользователей, как в графическом, так и в табличном виде, также её можно было бы получить за любой период, не тратя при этом много времени. Система позволила бы ускорить назначение управляющего дома, путем нажатия одной кнопки. Более того, сами пользователи могли бы отслеживать собственное потребление ежедневно, а не получать раз в месяц квитанцию с данными. Также они могли бы анализировать трату ресурсов благодаря графическому и табличному представлению данных в системе.

Таким образом, мы видим, что в сфере потребления питьевой воды, существует необходимость в автоматизированной системе.

## **Обзор литературы**

Обратимся к существующей на сегодняшний день литературе по теме мониторинга и анализа данных о продажах питьевой воды. Разобьем эту большую тему на несколько маленьких, разберемся в сфере продаж, возможности и способы анализа данных продаж, разберемся, для чего необходимы информационные системы в бизнесе.

Требуется разобраться в технологии проведения продаж, какие данные учитываются при составлении отчетов для отображения статистики. Каким образом, ведутся подсчёты, какие данные нужно знать, а какие получать на выходе.

Для начала разберем возможности и способы анализа данных продаж, чтобы лучше понимать какие системы требуются для автоматизации. В статье «Как проводить анализ продаж: этапы, методы и способы» [1], а также в книге «Практика продаж» Шнаппауфа Рудольфа [2] были выделены четыре вида анализа:

1. Анализ динамики объема продаж, задачей которого является определение изменения объема продаж предприятия по сравнению с предыдущим периодом.
2. Структурное исследование продаж, однако, данный анализ не подходит для предприятий с продажей одного вида товара.
3. Контрольный анализ объема продаж, делается в целях сравнения планируемых показателей с фактическими, требуется для своевременного принятия корректирующих действий.
4. Факторный анализ, благодаря которому можно определить факторы внутренней и внешней среды организации, которые повлияли на показатель оценки.

В следующей статье «Анализ продаж» [3] был выделен один из основных методов для проведения анализа продаж – ABC анализ.

Этапы проведения ABC-анализа номенклатуры товаров и объема продаж компании (предприятия) следующие:

1. Определение номенклатуры продукции предприятия.

2. Расчет нормы прибыли по каждой товарной группе.
3. Определение эффективности каждой группы.
4. Ранжирование товаров и их классификация (ABC) по ценности для предприятия.

Обращаясь к литературе, хотелось найти ответ на вопрос – как информационные системы помогают в бизнесе. В статье Джорджа Бойкина подчеркивается, что информационная система должна относиться к одной из трех категорий: периодическая, мониторинговая и запрашиваемая. Разработка маркетинговой информационной системы делает рабочий процесс проще и быстрее, она сортирует и анализирует данные абсолютно корректно, что может быть сложно решить вручную. В целом, концепция маркетинговых информационных систем - это быстрая обработка данных из новых источников, а также привязка внутренних данных к базам данных, ориентированным на потребителя, и принятие решения в соответствии с потребностями и бюджетом компаний [4].

Более того, чтобы реализовать информационную систему по анализу, мы должны знать, как рассчитать данные, какую формулу мы могли бы использовать, чтобы иметь правильные результаты. Как рассчитывать значения по кварталам, годам, дням и другим промежуткам, как выводить аналитику по значениям. Поэтому мы используем информацию из книги управленческого учета [5].

Также было выявлено, в статье «ABC АНАЛИЗ ПРОДАЖ. ПРИМЕР РАСЧЕТА В EXCEL», что основным инструментом для проведения расчётов данных является Excel, так он обладает следующими функциями [6]:

1. Обеспечивает быстрый поиск информации, достаточно занести соответствующие данные в программу.
2. Автоматически осуществляет расчёты.
3. Упрощает процесс анализа результатов, визуализируя их в виде диаграмм (особенно полезно при проведении контрольного анализа и анализа динамики объема продаж).

Таким образом, мы провели анализ источников, который помог глубже понять тему нашей выпускной квалификационной работы – какие методы анализа и мониторинга данных используются на сегодняшний день, и как мы можем применить данные знания на нашей работе.

## **Обзор существующих решений**

Требуется провести обзор существующих решений для выделения преимуществ и недостатков. Существует большое число систем учета воды, наиболее близкой по функциональному обеспечению является система «eMeterEnergy» (рис 1.1). Данный продукт допускает возможность слежения за всеми энергоресурсами, включая воду. Пользователь системы избавлен от содержания IT специалистов для обслуживания программных и технических средств системы, программные сервисы являются облачными (интернет), обслуживание технических средств входит в программу сервисного обслуживания. Существует возможность получения e-mail и sms сообщений. Несмотря на ряд преимуществ, данная система не в полной мере подходит для решения поставленной задачи, так как не обеспечивает заданной точности измерений – питьевая вода измеряется в литрах, нет возможности печати платежных квитанций, нет блоков математических расчетов.

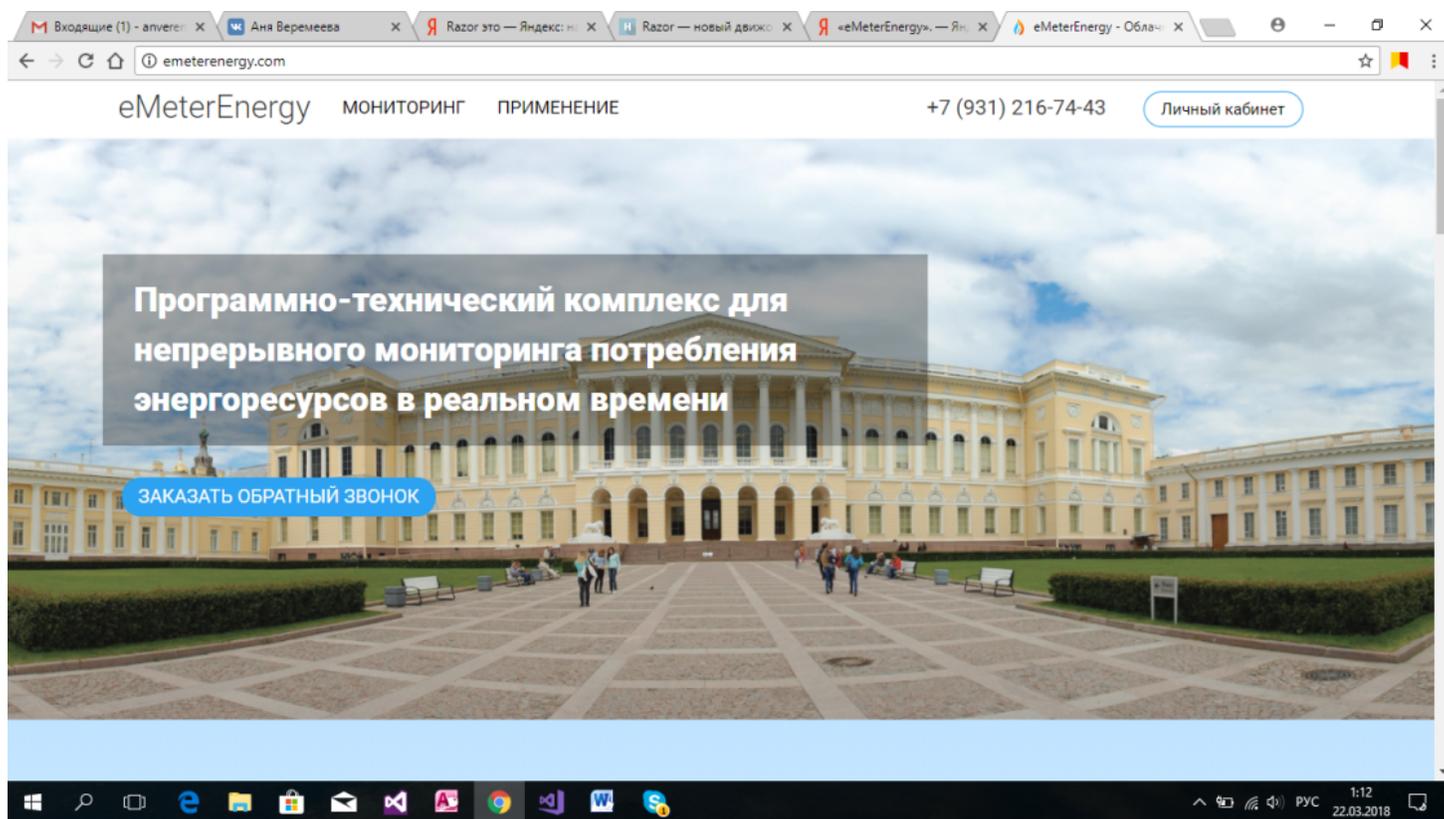


Рисунок 1.1. «eMeterEnergy»

Существует также система ГИС ЖКХ (рис. 1.2) – единая система Российской Федерации по учету всех жилищно-коммунальных услуг в домах, не только потребление воды. Это крупная информационная система государственного значения, но она не предусматривает работу по мониторингу и анализу такого ресурса, как питьевая вода.

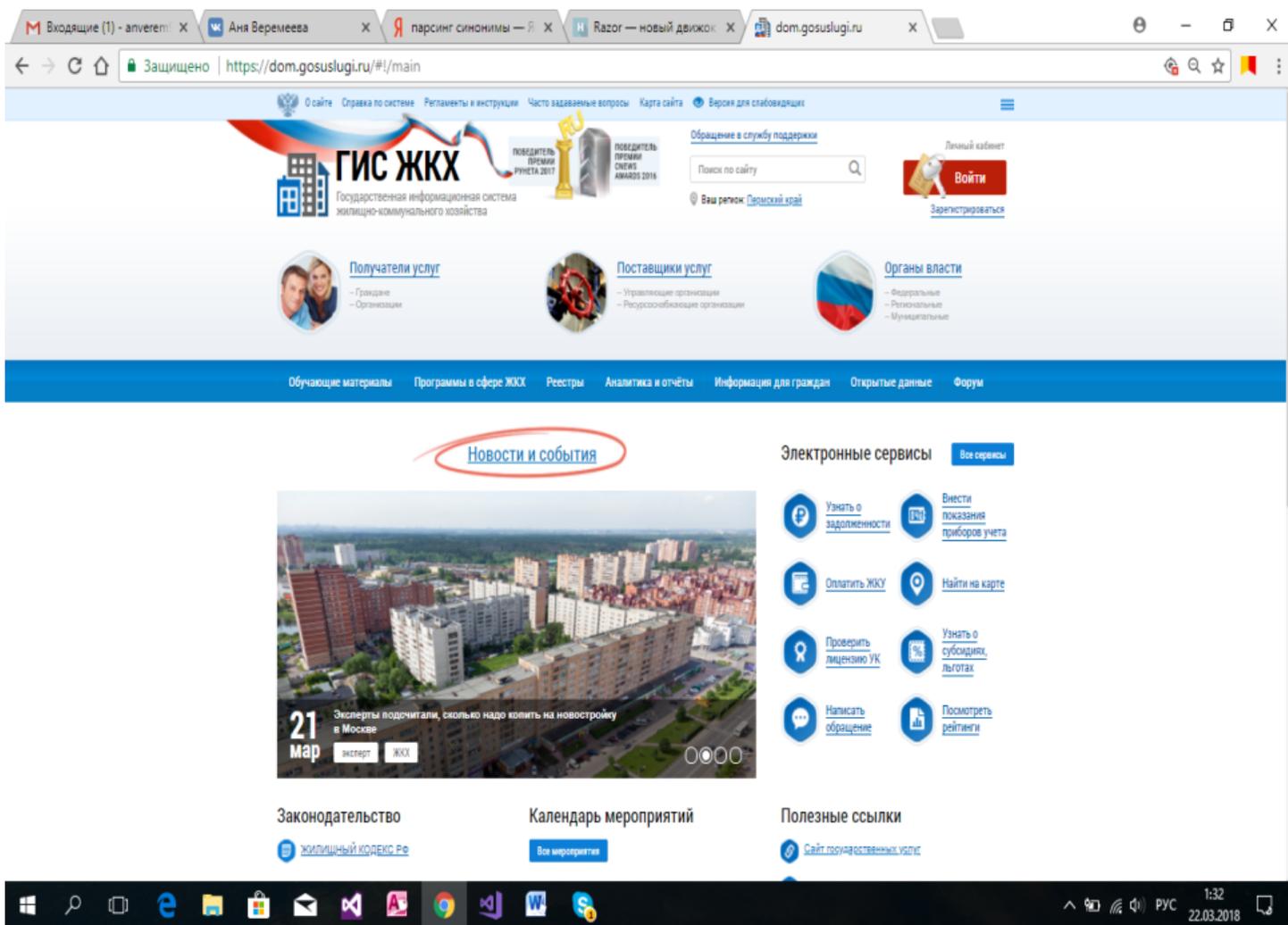


Рисунок 1.2. ГИС ЖКХ

Следующее существующее решение, которое мы рассмотрим, называется «Стриж» (рис.1.3), система беспроводного учета энергоресурсов в многоквартирных домах, жилых кварталах, поселках и городах, использующая беспроводную LPWAN-технология. Она позволяет производить онлайн учет электричества, воды, газа, тепла, предоставляет сбор показаний приборов в режиме реального времени через интернет без участия жильцов, без обходчиков. Однако данное решение также как и предыдущее, не позволяет производить учет именно питьевой воды, а также сам собственник не может пользоваться им, система предназначена только для управления для удобного сбора показаний.

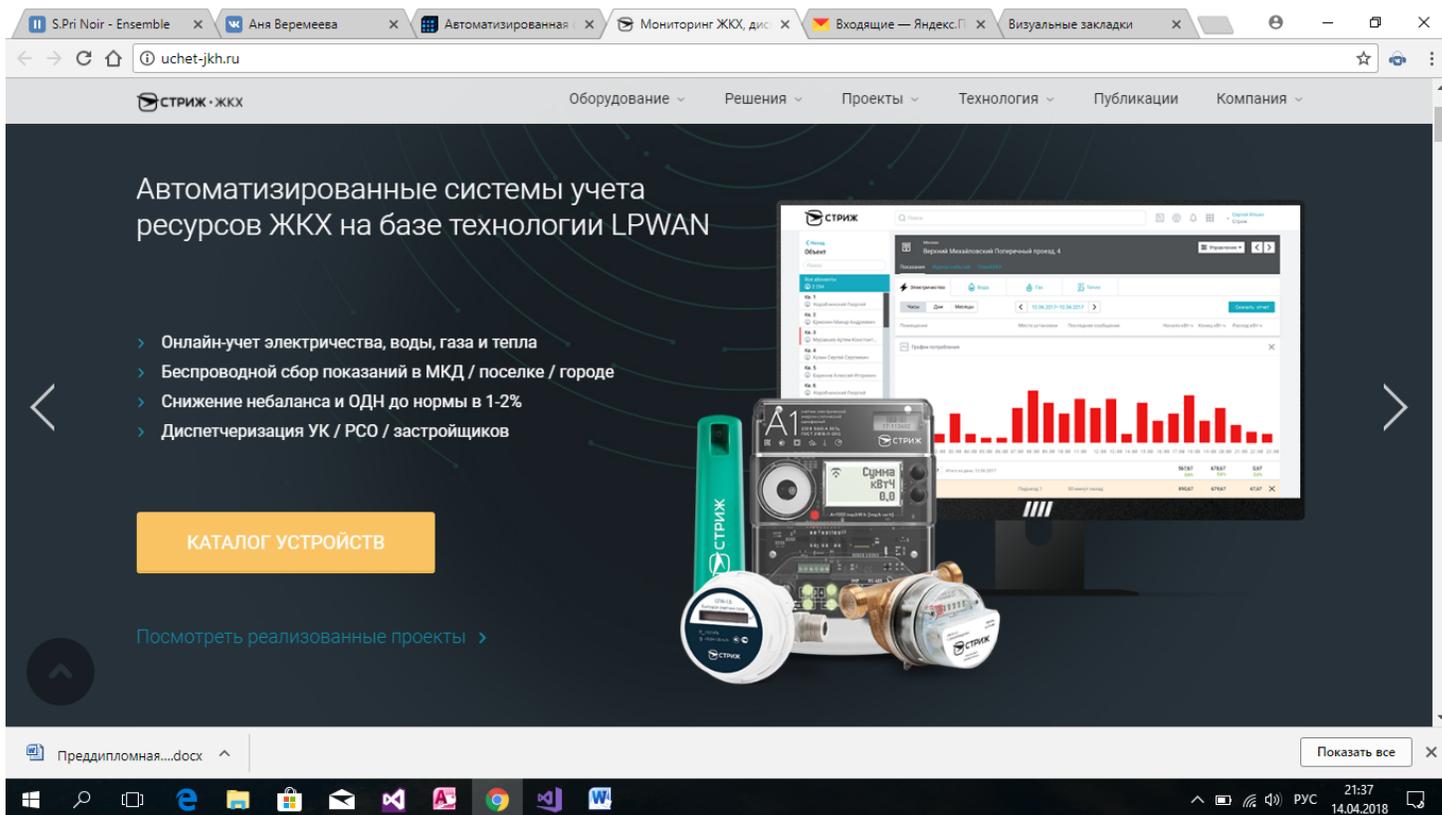


Рисунок 1.3. Система «Стриж»

На данный момент в Новороссийске активно разрабатывается автоматизированная система коммерческого отпуска питьевой воды потребителям. Заказчиком выступает комитет по управлению Жилищно-коммунальным хозяйством Администрации г. Новороссийск, а разработчиками являются - ЗАО «Взлет», ООО «Взлет-Кубань». Опишем состав функций данного решения:

- обеспечение приборного учета потребляемой питьевой воды каждым объектом водопотребления, отдельными потребителями с учетом принадлежащих им объектов
- контроль балансов водоснабжения с возможностью распределения потерь по технологическим группам (транспортировка – потребление) и местам возникновения затрат
- выявление источников неучтенных расходов и скрытых потерь и внедрение системы активного поиска утечек
- контроль лимитов питьевой воды (ПВ)
- поддержка принятия эффективных управленческих решений по вопросам подачи питьевой воды потребителям
- определение стратегии адресных, оптимально эффективных инвестиций, необходимых для модернизации и развития системы водоснабжения

Однако все описанные выше функции говорят о том, что система разрабатывается лишь для управленческого отдела, но не будет предоставлять информацию непосредственно потребителям, что является главным недочетом.

Стоит не забывать, что самостоятельный обход по квартирам и сбор данных, тоже является в той или иной степени существующим решением проблемы. Но сразу же можно выявить ряд минусов – требуется специальный человек, который будет совершать сбор данных по квартирам, а также выделение средств на его работу, дальнейшее занесение данных в базу будет осуществляться вручную.

Также существует ряд аналогичных систем направленных на слежение систем в доме, а нас интересует конкретно питьевая вода, так как учет питьевой воды не попадает под цели существующих решений, это новая ветка сервисов.

Приведем сравнительную таблицу рассмотренных решений (табл 1.1).

Таблица 1.1. Сравнительная таблица

<b>Система / Характеристика</b>	<b>EMeterEnergy</b>	<b>ГИС ЖКХ</b>	<b>Стриж</b>	<b>Сист. Новороссийск</b>	<b>Собственноручно</b>
Учет питьевой воды	+	-	-	+	+
Анализ данных	+	-	+	+	-
Доступ собственника квартиры к данным	-	+	-	-	-
Доступ управления к данным	+	+	+	+	+
Просмотр данных за любой период	+	+	+	+	-

Система / Характеристика	EMeterEnergy	ГИС ЖКХ	Стриж	Сист. Новороссийск	Собственноручно
Редактирование данных собственников	+	+	+	+	+
Просмотр тарифов потребления	-	+	-	+	+
Просмотр домов в системе	-	+	-	-	+
Просмотр пользователей в определенном доме	-	+	-	-	-

## Описание главных процессов информационной системы

Выделим основные процессы, которые мы будем разрабатывать в нашей системе.

Процесс: управление и слежение за потреблением питьевой воды во всех домах-клиентах.

Описание: Администратор имеет доступ ко всем функциям, он входит в систему по специальному логину и паролю, который ему предоставляет разработчик системы. Разработанная система должна будет предоставлять выполнять следующее: создавать и редактировать управляющих, создавать и редактировать дома и привязывать к определенному дому своего управляющего, добавлять и изменять тарифы на потребление питьевой воды. Создавать, изменять владельцев квартир, также он должен иметь возможность получить информацию о потреблении любого пользователя, с которым заключен договор об оказании услуг.

Интерфейс должен быть понятен пользователям и легко используемым. Каждая кнопка должна выполнять ту функцию, за которую она отвечает и ничего больше.

Каждая кнопка должна быть хорошо видна пользователю и характеризовать своё предназначение. Вид должен быть представлен в виде таблицы, для введения требуемой информации. На каждое действие пользователя должна быть ответная реакция системы.

Функции, которые должны быть реализованы, чтобы обеспечить выполнение описанного процесса:

1. Авторизация по логину и паролю.
2. Просмотр учётных записей.
3. Редактирование/удаление учётных записей.
4. Просмотр информации по тарифам.
5. Редактирование/удаление тарифов.
6. Просмотр домов.
7. Редактирование/удаление домов.
8. Просмотр анализа данных в табличном виде.
9. Просмотр анализа данных в графическом виде.

Процесс: получение информации о потреблении питьевой воды в доме.

Описание: Управляющий входит в систему по предоставленному администратором логину и паролю. Благодаря системе управляющий получает в короткие сроки данные о потреблении питьевой воды по всем собственникам квартиры, в домах под его руководством. В его правах заключать договора с новыми собственниками квартир и заносить их данные в систему.

Интерфейс должен быть понятен пользователям и легко используемым. Каждая кнопка должна выполнять ту функцию, за которую она отвечает и ничего больше. Каждая кнопка должна быть хорошо видна пользователю и характеризовать своё предназначение. Вид должен быть представлен в виде таблицы, для введения требуемой информации. На каждое действие пользователя должна быть ответная реакция системы.

Функции, которые должны быть реализованы, чтобы обеспечить выполнение описанного процесса:

- ○ ■
  1. Авторизация по логину и паролю.
  2. Создание нового собственника квартиры.
  3. Редактирование/ удаление собственника квартиры.
  4. Просмотр анализа данных по потреблению в табличном виде.

## 5. Просмотр анализа данных по потреблению в графическом виде.

Процесс: получение анализа данных по потреблению питьевой воды в собственной квартире.

Описание: Собственник квартиры входит в систему по предоставленному ему логину и паролю от управляющего, перед ним открывается форма, куда он вводит интересующий его интервал времени. За этот интервал пользователю предоставляется следующая информация: количество использованной воды и итоговая сумма, далее следует таблица, где за каждый день указано показания счётчика, расход и сумма за день, также вся информация по расходу предоставлена на графике.

Интерфейс должен быть понятен пользователям и легко используемым. Каждая кнопка должна выполнять ту функцию, за которую она отвечает и ничего больше. Каждая кнопка должна быть хорошо видна пользователю и характеризовать своё предназначение. Вид должен быть представлен в виде таблицы, для введения требуемой информации. На каждое действие пользователя должна быть ответная реакция системы.

Функции, которые должны быть реализованы, чтобы обеспечить выполнение описанного процесса:

1. Авторизация по логину и паролю.
2. Ввод временного промежутка.
3. Просмотр анализа данных в табличном виде.
4. Просмотр анализа данных в графическом виде.

### **Описание требований к информационной системе мониторинга и анализа данных о продажах питьевой воды**

Функциональным назначением программного продукта является осуществление сбора данных с предоставленных заказчиком серверов, возможность работы с ними пользователем, предоставление анализа системой затем графическое и табличное представление на экране, что позволит лучше проанализировать данные по потреблению.

Информационная система должна иметь удобный интерфейс для пользования, который не будет требовать специальных знаний от пользователей.

Система должна предоставлять следующие функции:

1. Различный доступ (собственник квартиры, управляющий, администратор). Собственник квартиры может просматривать данные только по своей квартире. Управляющий имеет доступ к определенным домам, а соответственно и к данным по всем квартирам в них, более того он создает в системе записи о новых собственниках квартир. Администратор обладает полным доступом – ко всем домам, ко всем пользователям, также он имеет возможность создавать новых управляющих. При входе система запрашивает логин и пароль, на основе которых предоставляет доступ к функциональности системы.
2. Сбор данных. Данные по потреблению питьевой воды хранятся на ftp-серверах, должен быть написан скрипт, который каждую ночь автоматически загружает все файлы с данными и переносит их в базу данных, которые затем переносятся на экран для просмотра пользователями.
3. Добавление новых данных. Система должна предоставлять возможность администратору и управляющему добавлять новые данные. Администратору – добавлять новые тарифы, новые дома, новых управляющих. Управляющему – добавлять в систему новых собственников квартир, с которыми компания заключила договор по предоставлению питьевой воды.
4. Редактирование данных. Система должна предоставлять возможность администратору и управляющему редактировать данные. Администратору – редактировать тарифы, дома, управляющих. Управляющему – редактировать в системе информацию о собственниках квартир, с которыми компания заключила договор по предоставлению питьевой воды.
5. Анализ данных. Требуется выводить статистику по потреблению, – какое количество было употреблено в определенный день, какой счетчик это показал, какая стоимость расхода. Пользователь может вводить любой промежуток и система должна вывести данные по нему в виде таблицы, а также в виде графика.
6. Удаление данных. Система должна предоставлять возможность администратору и управляющему удалять данные. Администратору – удалять тарифы, дома, управляющих. Управляющему – удалять в системе информацию о собственниках квартир, с которыми компания прервала договор по предоставлению питьевой воды.

Информационная система должна быть разработана в виде веб-сайта, это позволит иметь доступ к ней с любой точки при наличии интернета. Конечный продукт должен быть представлен для пользования в виде рабочей интернет-ссылки, которая позволит работать не только в локальной сети.

Для разметки веб-сайта использовать язык html [7]. Реализация веб-сайта должна быть осуществлена на языке php[8] в среде Phpdesigner. Так как нам требуется хранить данные с серверов в базе данных, требуется использовать MySQL для реализации данного действия.

Остальные требования к разрабатываемой системе определены в техническом задании (см. Приложение А).

### Определение трудоемкости разработки программы

Для определения трудоемкости нашей информационной системы необходимо составить список всех предстоящих основных видов работ, которые должны быть выполнены при разработке программного продукта. Эти работы объединяются в несколько этапов. При составлении программного продукта будут рассмотрены следующие этапы:

1. Составление технического задания.
2. Разработка архитектуры веб-сайта.
3. Разработка интерфейса веб-сайта.
4. Реализация информационной системы.
5. Тестирование работоспособности веб-сайта.

По каждому этапу указываются их конкретные исполнители, так как всеми этапами занимается единственный человек, у нас он один - студент.

Трудоемкость выполнения разработки программного обеспечения определяется по сумме трудоемкости этапов разработки и видов работ, оцениваемых экспертным путем в человеко-днях (таблица 1.2)

**Таблица 1.2. Расчет трудоемкости разработки программы**

№	Вид работы	Исполнитель	Оценка трудоемкости (чел/дн)			Т ожидаемое (чел/дн)
			2	3	4	
1	Составление ТЗ	Студент	2	3	4	3

**Оценка  
трудоемкости  
(чел/дн)**

№	Вид работы	Исполнитель	Оценка трудоемкости (чел/дн)			Т ожидаемое (чел/дн)
			Изучение задания, сбор исходных данных	Студент		
2			2			1
3	Анализ имеющихся данных	Студент	5	6	7	6
4	Обзор имеющихся решений	Студент	7	7	8	7
5	Разработка архитектуры	Студент	5	5	6	5
6	Разработка алгоритма	Студент	2	3	4	3
7	Разработка интерфейса веб-сайта	Студент	6	7	9	7
8	Разработка веб-сайта	Студент	50	60	70	60
9	Отладка и тестирование	Студент	4	5	6	5
10	Оценка полноты решения поставленной задачи	Студент	1	2	3	2
Итого:						100

Для определения ожидаемого времени, используем следующую формулу:

(1)

где  $T_{\min}$  – минимально возможная трудоемкость;

$T_{\max}$  – максимально возможная трудоемкость;

$T_{\text{нв}}$  – наиболее вероятная трудоемкость.

Таким образом, после всех подсчетов и расстановки времени, мы рассчитали ожидаемое время, требующееся для разработки нашего программного продукта, с помощью формулы (1), и оно составило 100 чел/дн.

## ГЛАВА 2. ПРОЕКТИРОВАНИЕ ЭКОНОМИЧЕСКОЦ ИНФОРМАЦИОННОЙ СИСТЕМЫ МОНИТОРИНГА И АНАЛИЗА ДАННЫХ О ПРОДАЖАХ ПИТЬЕВОЙ ВОДЫ

### Диаграмма прецедентов

Используем в своей работе диаграмму прецедентов (рис.2.1), которая отражает отношения между акторами в нашем случае их три – администратор, управляющий и собственник, и прецедентами. Она позволит нам дать описание системы на концептуальном уровне. Основным назначением диаграммы является описание её поведения, функциональности, она упрощает понимание системы одновременно для заказчика, разработчика и пользователя. Смотри на диаграмму можно без труда распознать актора и его взаимодействие с системой. На нашей диаграмме (рис.2.1) каждый прецедент имеет своего инициатора, в нашем случае их три – «Администратор», «Управляющий», «Собственник» и приводит к соответствующему результату.

На диаграмме видно, что больше всех прав у администратора, ему доступны все функции – создание, редактирование, удаление управляющего, собственника, дома. Он также как и все авторизуется в системе. Он может получить статистику по потреблению любого пользователя, зарегистрированного в системе.

Управляющий имеет ограниченный доступ – он может создавать, удалять, редактировать собственников, того дома, которым управляет и просматривать статистику по потреблению.

Собственник обладает самым малым перечнем функций – он может просматривать статистику по потреблению только в своей квартире.

Администратор

Управляющий

«Включает»

«Включает»

«Включает»

«Включает»



Система «Анализ и мониторинг данных по продаже воды»

*Рисунок 2.1. Диаграмма прецедентов*



Собственник

Прецеденты информационной системы

Прецедент – описание множества содержательно-близких сценариев взаимодействия акторов (внешних агентов) с системой, которое осуществляется с целью получения акторами некоторого полезного результата при помощи системы. Как нам это поможет в реализации информационного приложения? Процесс выделения акторов тесно связан с определением границ системы. Когда мы определим границы нашей информационной системы, мы сможем определить её внешние и внутренние свойства. Внешняя среда представляется только акторами. В простейшем случае прецедент представляет собой один или несколько абзацев текста, описывающих взаимодействие акторов с самой системой. Однако гораздо удобнее и эффективнее использовать шаблоны, состоящие из набора полей. Рассмотрим прецеденты, которые мы выделили:

1. Авторизация в приложении (табл.2.1.)

Актеры: администратор, управляющий, собственник.

Краткое описание: пользователь входит в систему, вводит в специальную форму свой логин и пароль, которые он получил от вышестоящих ролей (собственник от управляющего, управляющий от администратора, администратор от разработчика системы) и входит в систему, в зависимости от своего доступа видит соответствующие вкладки:

- 1. Администратор – полный доступ, возможность просмотра всех данных по всем домам и квартирам, создание и редактирование управляющих и собственников.
- 2. Управляющий – ограниченный доступ, возможность создания собственников квартиры, просмотр потребления по частичному дому, создание договоров.
- 3. Собственник – ограниченный доступ, возможность просмотра графика потребления по своей квартире.

S1 Основной поток:

Таблица 2.1. Авторизация в приложении

<b>Действия акторов</b>	<b>Отклик системы</b>
<b>1. Нажимает на вход в систему</b>	<b>1. Открывает форму для заполнения данных</b>
<b>1. Заполняет соответствующие поля (логин и пароль)</b>	<b>1. Заполненная форма</b>
<b>1. Нажимает «вход»</b>	<b>1. Вывод вкладок, соответствующих роли пользователя</b>

E1 Подпоток:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Появление ошибки при отсутствии данных в одном из полей. Следует нажатие на кнопку «Ок». Пользователь предотвращает ошибку, вводя недостающие данные или же нажатием кнопки «Отмена». Прецедент продолжается.

#### 1. Создание управляющего (табл. 2.2.)

Актеры: администратор

Краткое описание: администратор находится в системе, он переходит на вкладку «учетные записи», где ему доступны операции над пользователями с ролью «управляющий», он нажимает на добавление нового пользователя, вводит соответствующие данные: ФИО, e-mail, пароль, телефон и создает новый профиль.

S1Основной поток:

Таблица 2.2. Создание управляющего

<b>Действия акторов</b>	<b>Отклик системы</b>
1. Входит в систему	1. Вкладки для пользователя «администратор»
1. Нажимает на вкладку «учетные записи»	1. Открывается форма с информацией о пользователях с ролью «управляющий»
1. Нажимает добавить учетную запись	1. Открывается специальная форма
1. Вводит необходимые данные: фио, e-mail, пароль, телефон	1. Заполненная форма
1. Нажимает кнопку «Добавить пользователя»	1. Сохранение данных в базу

E1Подпоток:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Появление ошибки при не заполненном данными поле и последующим нажатием кнопки «Ок». Пользователь предотвращает ошибку, вводя недостающие данные или же нажатием кнопки «Отмена». Прецедент продолжается.

1. Редактирование управляющего (табл. 2.3.)

Актеры: администратор

Краткое описание: администратор находится в системе, он переходит на вкладку «учётные записи», где ему доступны операции над пользователями с ролью «управляющий», он нажимает на редактирование определенного пользователя, корректирует соответствующие данные: ФИО, e-mail, пароль, телефон и сохраняет изменения.

S1Основной поток:

Таблица 2.3. Редактирование управляющего

<b>Действия акторов</b>	<b>Отклик системы</b>
1. Входит в систему	1. Вкладки для пользователя «администратор»
1. Нажимает на вкладку «управляющие»	1. Открывается форма с информацией о пользователях с ролью «управляющий»
1. Нажимает редактировать пользователя	1. Открывается специальная форма
1. Редактирует необходимые данные: фио, e-mail, пароль	1. Заполненная форма

## Действия акторов

## Отклик системы

1. Нажимает кнопку «Сохранить изменения»

1. Сохранение данных в базу

E1Подпотоки:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Появление ошибки при не заполненном данными поле и последующим нажатием кнопки «Ок». Пользователь предотвращает ошибку, вводя недостающие данные или же нажатием кнопки «Отмена». Прецедент продолжается.

1. Удаление управляющего (табл. 2.4.)

Акторы: администратор

Краткое описание: администратор находится в системе, он переходит на вкладку «учётные записи», где ему доступны операции над пользователями с ролью «управляющий», он удаляет пользователя из базы.

S1Основной поток:

Таблица 2.4. Удаление управляющего

## Действия акторов

## Отклик системы

1. Входит в систему

1. Вкладки для пользователя «администратор»

1. Нажимает на вкладку «учётные записи»

1. Открывается форма с информацией о пользователях с ролью «управляющий»

1. Нажимает удалить пользователя

1. Форма: Вы уверены, что хотите удалить эту запись?

## Действия акторов

## Отклик системы

1. Нажимает «Да»

1. Форма: Удаление прошло успешно; удалено из базы

E1Подпотоки:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Нажимает на кнопку «отмена» при всплывающем окне с вопросом об уверенности удаления строки. Удаление не происходит. Повторное выделение строки и нажатие кнопки удаления. Прецедент продолжается.

1. Создание дома (табл. 2.5.)

Акторы: администратор

Краткое описание: администратор находится в системе, он переходит на вкладку «дома», где ему доступны операции над домами, он нажимает на добавление нового дома, вводит соответствующие данные: адрес, количество квартир, количество этажей и назначает управляющего после чего осуществляет добавление новых данных в базу.

S1Основной поток:

Таблица 2.5. Создание дома

## Действия акторов

## Отклик системы

1. Входит в систему

1. Вкладки для пользователя «администратор»

1. Нажимает на вкладку «дома»

1. Открывается форма с информацией о домах

## Действия акторов

## Отклик системы

1. Нажимает добавить дом

1. Открывается специальная форма

1. Вводит необходимые данные: адрес, количество квартир, количество этажей, управляющего

1. Заполненная форма

1. Нажимает кнопку «Добавить»

1. Сохранение данных в базу

E1Подпотоки:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Появление ошибки при не заполненном данными поле и последующим нажатием кнопки «Ок». Пользователь предотвращает ошибку, вводя недостающие данные или же нажатием кнопки «Отмена». Прецедент продолжается.

1. Редактирование дома (табл. 2.6.)

Акторы: администратор

Краткое описание: администратор находится в системе, он переходит на вкладку «дома», где ему доступны операции над домами, он нажимает на кнопку «редактировать» напротив желаемого объекта. Открывается форма с данными, куда администратор вносит изменения, после чего сохраняет новые данные.

S1Основной поток:

Таблица 2.6. Редактирование дома

## Действия акторов

## Отклик системы

1. Входит в систему	1. Вкладки для пользователя «администратор»
1. Нажимает на вкладку «дома»	1. Открывается форма с информацией о домах
1. Нажимает «редактировать» напротив желаемого дома	1. Открывается специальная форма
1. Корректирует имеющиеся данные	1. Измененная форма
1. Нажимает кнопку «Сохранить»	1. Сохранение данных в базу

E1Подпотоки:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Появление ошибки при не заполненном данными поле и последующим нажатием кнопки «Ок». Пользователь предотвращает ошибку, вводя недостающие данные или же нажатием кнопки «Отмена». Прецедент продолжается.

1. Удаление дома (табл. 2.7.)

Актеры: администратор

Краткое описание: администратор находится в системе, он переходит на вкладку «дома», где ему доступны операции над домами, он нажимает напротив желаемого объекта кнопку удалить, подтверждает свои действия, происходит удаление.

S1Основной поток:

Таблица 2.7. Удаление дома

**Действия акторов**

**Отклик системы**

1. Входит в систему	1. Вкладки для пользователя «администратор»
1. Нажимает на вкладку «дома»	1. Открывается форма с информацией о домах
1. Нажимает кнопку «удалить» напротив желаемого дома	1. Открывается специальная форма: Уверены ли вы, что хотите удалить?
1. Нажимает «да»	1. Удаление из базы

E1Подпотоки:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Появление ошибки при удалении дома, к которому еще прикреплены пользователи. Управляющий предотвращает ошибку, прикрепляя собственников к другому дому. Прецедент продолжается.

1. Создание собственника (табл. 2.8.)

Актеры: администратор, управляющий

Краткое описание: управляющий находится в системе, он переходит на соответствующую вкладку, где ему доступны операции над пользователями с ролью «собственник», он нажимает на добавление нового пользователя, вводит соответствующие данные: ФИО, e-mail, пароль, дом, квартира, создает новый профиль.

S1Основной поток:

Таблица 2.8. Создание собственника

**Действия акторов**

**Отклик системы**

- |  |  |
|--|--|
| 1. Входит в систему  | 1. Вкладки для пользователя «управляющий»                                |
| 1. Нажимает на вкладку «собственники»                            | 1. Открывается форма с информацией о пользователях с ролью «собственник» |
| 1. Нажимает добавить пользователя                                | 1. Открывается специальная форма   |
| 1. Вводит необходимые данные: фио, e-mail, пароль, дом, квартира | 1. Заполненная форма   |
| 1. Нажимает кнопку «Добавить пользователя»                       | 1. Сохранение данных в базу  |

E1Подпотоки:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Появление ошибки при не заполненном данными поле и последующим нажатием кнопки «Ок». Пользователь предотвращает ошибку, вводя недостающие данные или же нажатием кнопки «Отмена». Прецедент продолжается.

1. Редактирование собственника (табл. 2.9.)

Актеры: администратор, управляющий

Краткое описание: управляющий находится в системе, он переходит на соответствующую вкладку, где ему доступны операции надо пользователями с ролью «собственник», он нажимает на редактирование определенного пользователя, корректирует соответствующие данные: ФИО, e-mail, пароль, дом, квартира, сохраняет изменения.

S1Основной поток:

Таблица 2.9. Редактирование собственника

## Действия акторов

## Отклик системы

- |  |  |
|--|--|
| 1. Входит в систему                                    | 1. Вкладки для пользователя «управляющий»                                |
| 1. Нажимает на вкладку «собственники»                  | 1. Открывается форма с информацией о пользователях с ролью «собственник» |
| 1. Нажимает редактировать пользователя                 | 1. Открывается специальная форма   |
| 1. Редактирует необходимые данные: фио, e-mail, пароль | 1. Заполненная форма   |
| 1. Нажимает кнопку «Сохранить изменения»               | 1. Сохранение данных в базу  |

E1Подпотоки:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Появление ошибки при не заполненном данными поле и последующим нажатием кнопки «Ок». Пользователь предотвращает ошибку, вводя недостающие данные или же нажатием кнопки «Отмена». Прецедент продолжается.

1. Удаление собственника (табл. 2.10.)

Акторы: администратор, управляющий

Краткое описание: управляющий находится в системе, он переходит на соответствующую вкладку, где ему доступны операции надо пользователями с ролью «управляющий», он удаляет пользователя из базы.

S1Основной поток:

Таблица 2.10. Удаление собственника

<b>Действия акторов</b>	<b>Отклик системы</b>
1. Входит в систему	1. Вкладки для пользователя «управляющий»
1. Нажимает на вкладку «собственники»	1. Открывается форма с информацией о пользователях с ролью «собственник»
1. Нажимает удалить пользователя	1. Форма: Уверены ли вы, что хотите удалить ... собственника?
1. Нажимает «Да»	1. Форма: Удаление прошло успешно; удалено из базы

E1Подпотоки:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Нажимает на кнопку «отмена» при всплывающем окне с вопросом об уверенности удаления строки. Удаление не происходит. Повторное выделение строки и нажатие кнопки удаления. Прецедент продолжается.

1. Статистика по потреблению (табл. 2.11.)

Актеры: администратор, управляющий, собственник

Краткое описание: пользователь заходит в систему, переходит на вкладку «статистика по потреблению», вводит промежуток желаемого времени, система предоставляет статистику за указанный период в графическом и табличном виде.

S1 Основной поток:

Таблица 2.11. Статистика по потреблению

## Действия акторов

## Отклик системы

1. Пользователь переходит на вкладку «статистика по потреблению»

1. Вкладка «статистика по потреблению»

1. Вводит желаемые даты, нажимает ок

1. Вывод графика и таблицы по статистике за требуемый период

E1Подпотоки:

Запуск системы, происходит загрузка ранее введенных данных.

Альтернативные потоки:

Появление ошибки при нажатии на кнопку отображения в график и не введенных дат, предотвращение ошибки, ввод требуемых данных. Прецедент продолжается.

Архитектура информационной системы

В данном разделе производится разработка архитектуры системы, и описывается алгоритм работы будущего веб-сайта по осуществлению сбора и анализа данных по продажам питьевой воды.

Для лучшего понимания процесса в нашей будущей системе опишем её алгоритм:

1. Ежедневно с предоставленных ftp-серверов производится сбор требуемых файлов с данными.
2. Далее осуществляется разбор этих файлов с занесением в нашу базу данных по указанным ранее путям и обработка ошибок, файлы перезаписываются ежедневно.
3. В приложении пользователи имеют определенную роль – собственник квартиры, управляющий, администратор, все они имеют разный доступ к системе. У каждого есть логин и пароль, которые они указывают при входе в систему.
4. Система проверяет - существуют ли указанные данные в базе, если да, то переходим к следующему шагу, если нет, появляется сообщение об ошибке – «Неправильный логин или пароль». Если пользователь не может войти в систему, ему стоит обратиться к вышестоящему для исправления ошибки.

5. Осуществляется определение роли пользователя, при входе в систему.
  1. Роль – администратор? Если нет, проверяем на другую роль.
    1. Открываем на страницу с вкладками меню только для администратора.
      1. Управление данными по управляющим.
      2. Управление данными по собственникам.
      3. Управление данными по домам.
      4. Управление данными по тарифам.
      5. Управление данными по всем домам и квартирам.
    2. Роль – управляющий? Если нет, то роль собственника.
      1. Открываем страницу с вкладками меню только для управляющего.
        1. Управление данными по собственникам.
        2. Просмотр домов в управлении.
    3. Роль – собственник квартиры.
      1. Перенаправляем на страницу с вкладками меню только для собственника.
        1. Вывод данных в квартире собственника.

Рассмотрим используемые методы в работе некоторых функций. Наша система каждый день собирает файлы с сервера и сохраняет их. Наш разбор состоит из трёх этапов: сначала сбор информации в первоначальном виде. Автоматический разбор подразумевает копирование кода конкретной страницы с последующим извлечением из неё контента (необходимых данных). Затем наступает фаза извлечения и дальнейшего изменения формата информации. На этом этапе совершаем преобразование в требуемый формат. Генерация результата – это последняя фаза разбора. Во время неё записывается информация, полученная в процессе предыдущей фазы. И сразу же переводится в необходимый формат с последующей записью в локальную базу данных.

Создадим модель нашей базы данных, благодаря которой работа системы изнутри будет понятнее (рис.2.2.). Наша база данных имеет 6 сущностей – Houses, которая характеризует дома в системе, Controllers, хранит информацию о счётчиках, Data показания учета воды, Users, хранит информацию о пользователях, Rate, хранит информацию о тарифах потребления, Managers – хранит информацию об управляющих.

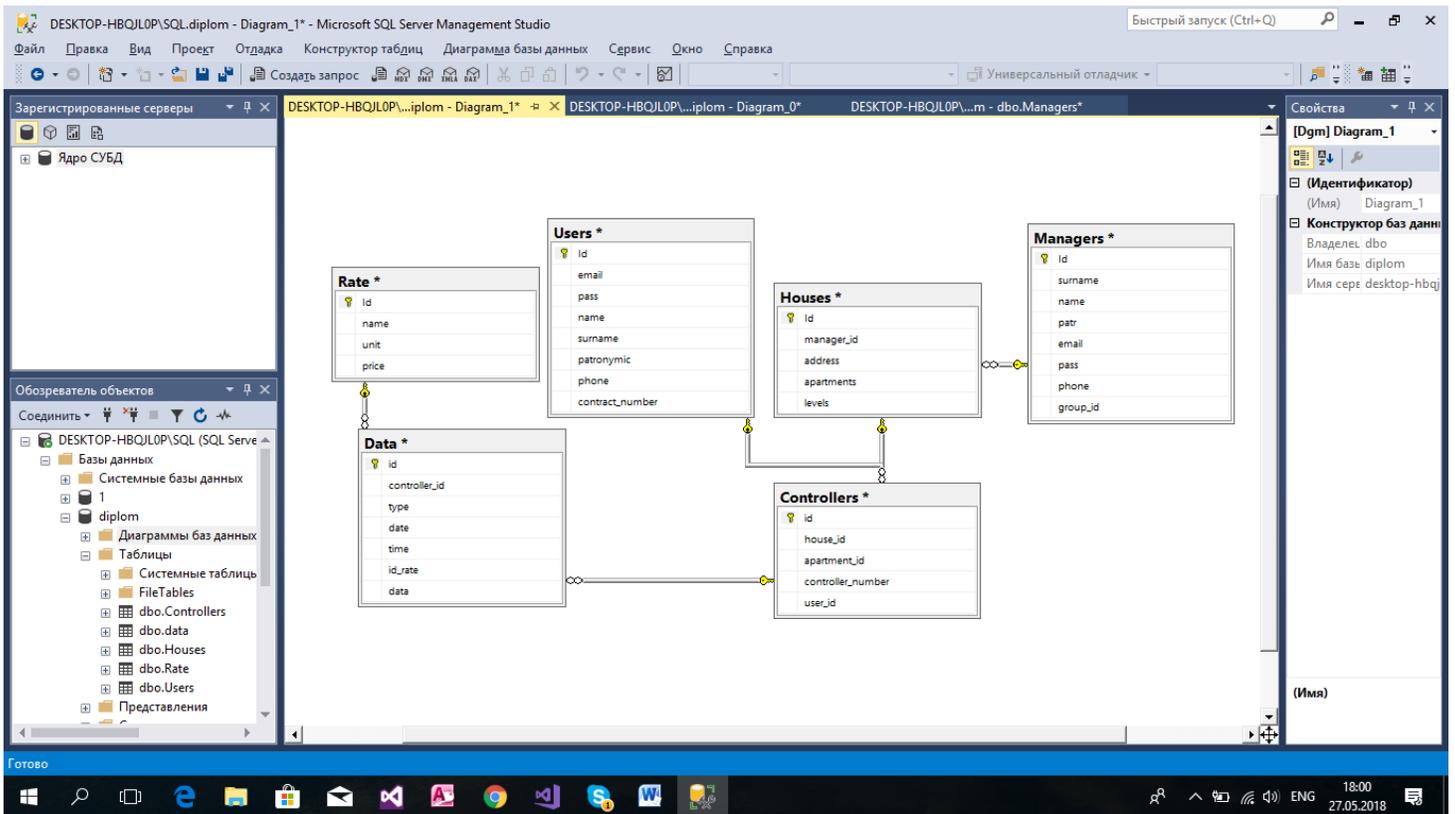


Рисунок 2.2. Модель базы данных

Разберем каждую сущность конкретнее:

1. Houses – собирает информацию о домах в нашей системе, имеет id уникальный идентификатор дома, manager\_id по которому осуществляется привязка к конкретному управляющему дома, address адрес, где расположен определенный дом, apartments несет информацию о количестве квартир и levels о количестве этажей.
2. Managers – собирает информацию об управляющем персонале в системе, имеет уникальный идентификатор id, surname, name, patr – фиио соответственно, email – логин, pass – пароль, phone – номер телефона, group\_id – идентификатор, по которому определяется управляющий или администратор.
3. Controllers – собирает информацию о счётчиках, имеет id уникальный идентификатор счётчика, house\_id, apartment\_id – несёт информацию о доме и квартире соответственно, где размещен счётчик, controller\_number – уникальный номер счётчика и user\_id – идентификатор, определяющий пользователя.
4. Data – собирает информацию о данных потребления в квартирах, id – уникальный идентификатор, controller\_id – несет информацию о счётчике, type – тип счетчика, date – дата, time – время, data – количество потребленных

литров воды, id\_rate – идентификатор, по которому определяется расчётный тариф.

5. Rate – собирает информацию по тарифу потребления, имеет уникальный идентификатор id, name – название тарифа, unit – мера, price – цена за единицу меры.
6. Users – собирает информацию о собственниках квартиры, id – уникальный идентификатор, e-mail – почта пользователя, одновременно является логином для входа в систему, pass – пароль для входа, name, surname, patronymic – фио соответственно, phone – номер телефона, contract\_number – номер договора.

Составим схему для описания архитектуры работы нашей будущей системы (рис. 2.3), которая хорошо показывает альтернативу связке Denwer-MySQL-PHP. Модули аккаунты (accounts), дома (houses), модуль главной страницы (main), настройки (settings), статистика (stat), тарифы (tariff), пользователи (users) вынесены как увеличивающие функциональность системы. В то же время система не зависит от них, и сами эти модули самодостаточны. Модуль авторизации вынесен в отдельную часть, поскольку он не реализует дополнительную функциональность системы, а является одной из ее частей.

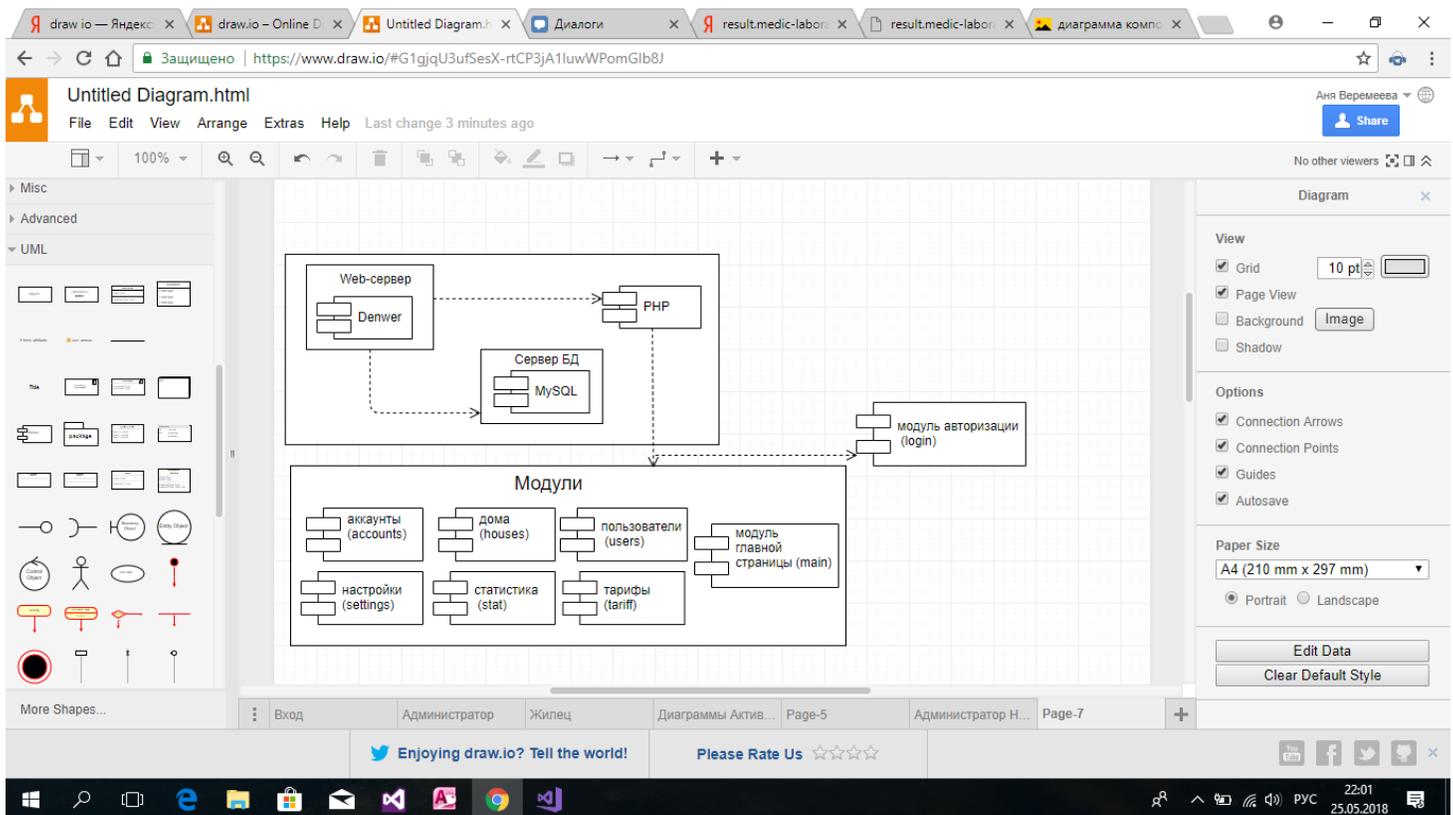


Рисунок 2.3. Архитектура информационной системы

Для описания особенностей физического представления нашего веб-сайта составим диаграмму компонентов (рис. 2.4). Она позволит описать архитектуру разрабатываемого сайта, установив зависимости с помощью пунктирных стрелок между компонентами, в роли которых выступят файлы с кодом.

`index.php` – главный скрипт, через который будет осуществляется вся работа.

`tmp.inc.php` - главный html шаблон, который повторяется на всех страницах сайта.

`config.php` - пароли от базы данных и от удаленного сервера, где лежат csv файлы.

`lib/functions.php` – файл, содержащий все функции, которые используются в коде.

`lib/db.php` – подключение к базе данных и функция для работы с запросами к базе данных.

`js/script.js` - набор средств, которые позволяют создавать более интерактивные веб-страницы.

`css/style.css` - папка в которой находятся файлы стилей (для оформления сайта).

- /название\_модуля/`tmp.inc.php` - html шаблон модуля.

- /название\_модуля/`index.php` - php код модуля, который будет нужен для вывода информации при открытии страницы данного модуля.

- /название\_модуля/`ajax.php` - код, который будет обрабатывать асинхронные запросы (все действия, которые будут совершаться без перезагрузки страницы, например, добавить или удалить пользователя).

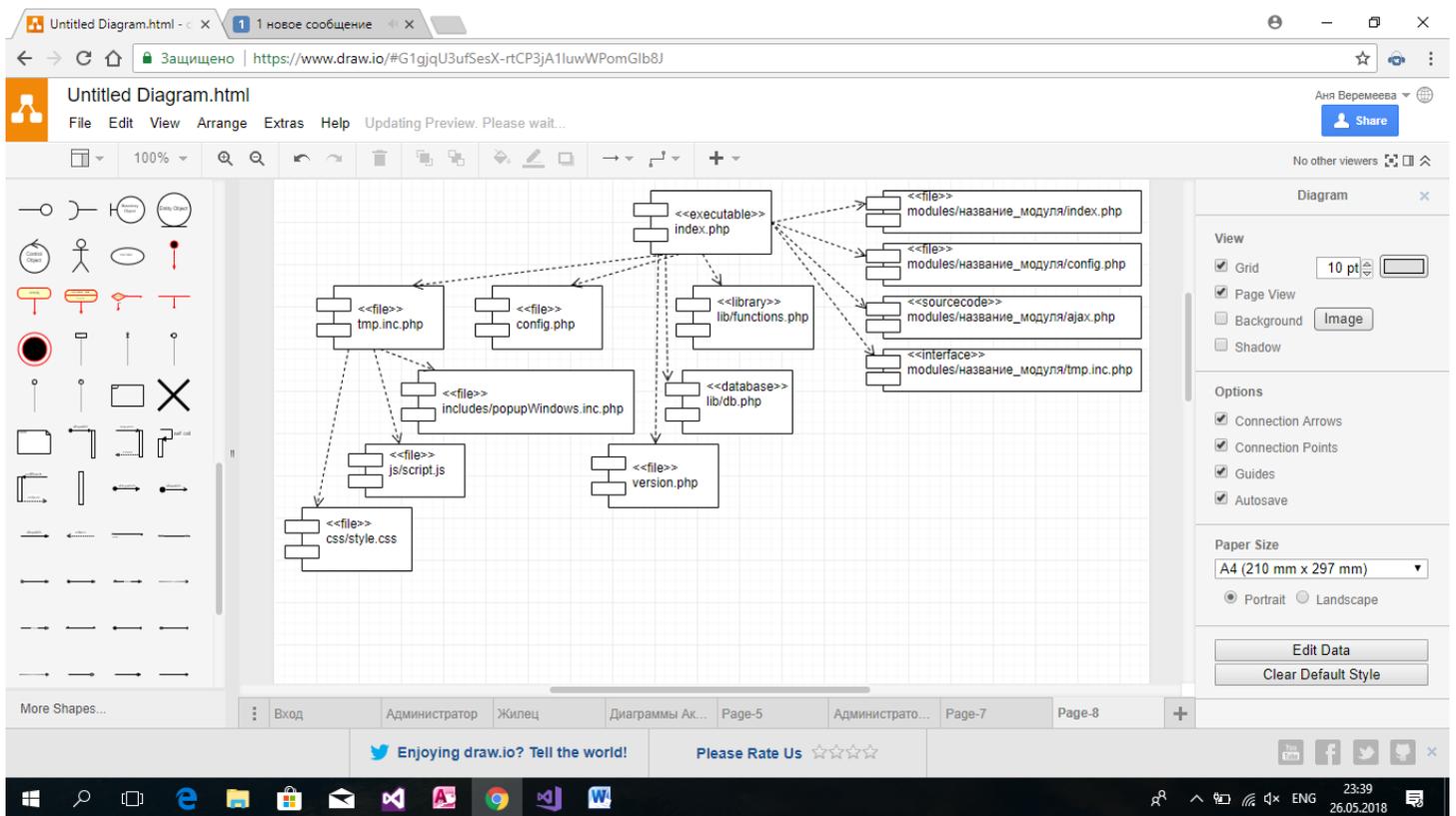


Рисунок 2.4. Диаграмма компонентов

### ГЛАВА 3. РЕАЛИЗАЦИЯ ЭКОНОМИЧЕСКОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ МОНИТОРИНГА И АНАЛИЗА ДАННЫХ О ПРОДАЖАХ ПИТЬЕВОЙ ВОДЫ

Разработка информационной системы мониторинга и анализа данных о продажах питьевой воды

Разрабатываемая информационная система направлена на мониторинг и анализ данных о продажах питьевой воды и представляет собой комплекс элементов, которые связаны между собой. Пользователь осуществляет действия в системе через браузер при наличии доступа к сети Интернет, с помощью любого устройства. Все операции будут выполняться на веб-сервере, который также будет работать с сервером баз данных, который позволит нам хранить загружаемые данные о потреблении с ftp-серверов. Схема работы информационной системы представлена на рисунке 3.1.

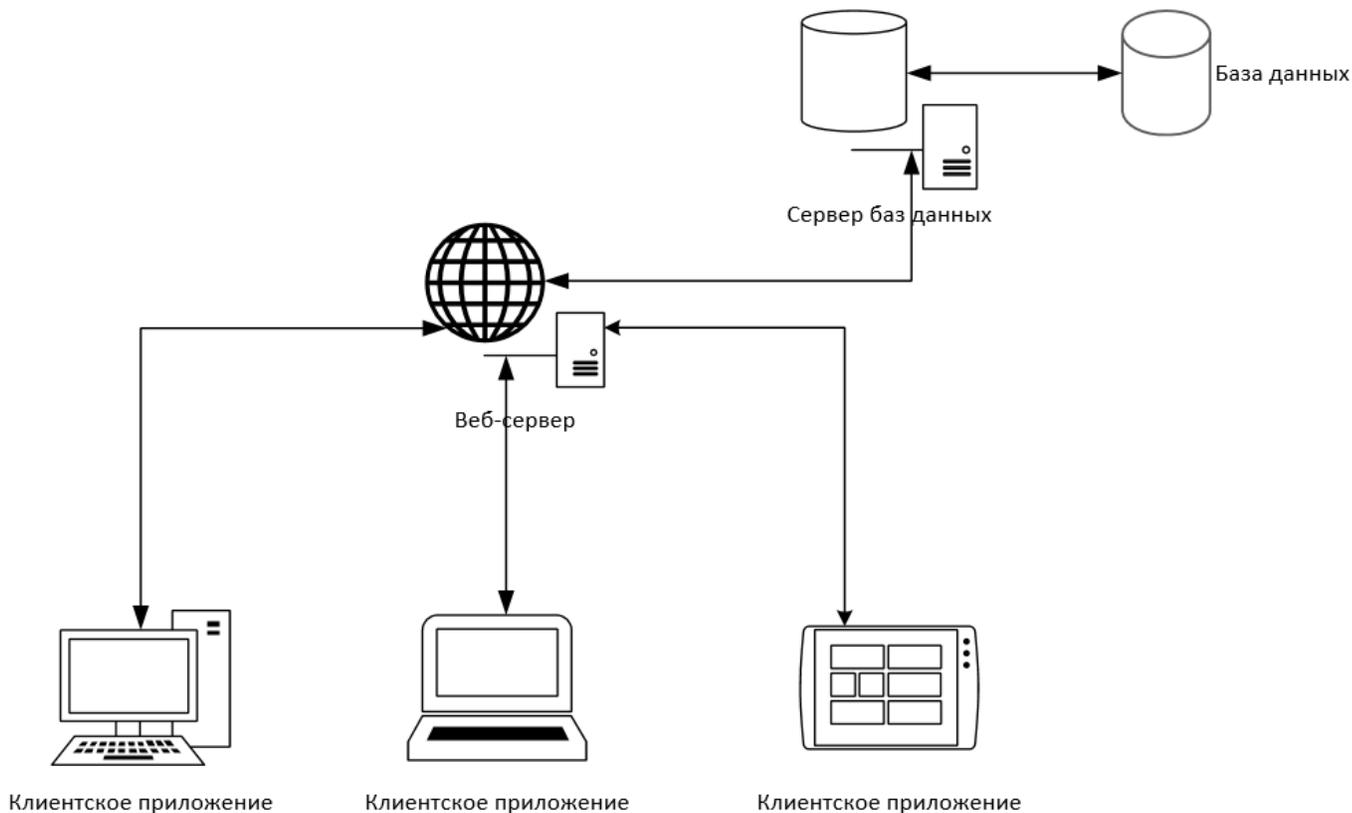


Рисунок 3.1. Схема работы информационной системы

Для лучшего понимания структуры работы веб-сайта, составим схему последовательности открытия окон (рис.3.2). Сначала, пользователь должен будет внести свои данные для входа, далее форма зависит от прав пользователя, администратор – полный доступ, управляющий доступ к функциям по управлению домом, собственник – доступ к собственным данным.

Вход

Панель для собственника

Панель для управляющего

Панель для администратора

Пользователи

Личное потребление

Дома

Дома

Тарифы

Настройки

Пользователи

Настройки

Учётные записи

Настройки

*Рисунок 3.2. Схема последовательности открытия окон*

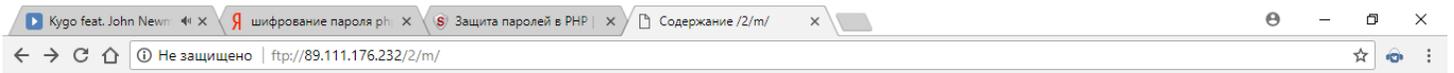
При входе в систему пользователь вводит логин и пароль на основе которых получает доступ к тем или иным данным. Если хранить все пароли в первичном виде, то это может привести к плохим последствиям, если кто-то завладеет базой данных, поэтому пароли принято шифровать. Было принято решение осуществлять данное преобразование с помощью хеш функций, что означает преобразование пароля в случайную строку, из которой будет достаточно сложно узнать оригинальный пароль.

После введения пароля, первоначально он обрабатывается функцией md5, после чего получается длинная строка из латинских букв и цифр. Далее эта строка поддается обработке функцией sha1, что приводит к получению новой строки из букв и цифр. Затем к полученной строке присоединяется тайная строка, после чего вновь обрабатываем функциями криптошифрования md5 и sha1.

Кодшифрования:

```
$pass = sha1(md5(trim($pass)));  
$pass = md5($pass . 'srrfrwfrfesfwfsddddjkl');
```

В нашей работе мы используем данные на ftp-серверах (рис.3.3), для быстрого отклика системы, был написан скрипт, который запускается раз в сутки ночью (наименьшая активность пользователей), скачивает csv файлы с удалённого сервера и записывает показания из этих файлов в базу.



## Содержание /2/m/

[родительский каталог]

Название	Размер	Последнее изменение
00000375.csv	440 B	21.04.2018, 20:47:00
00000381.csv	302 B	21.04.2018, 20:53:00
00000413.csv	1.3 kB	21.04.2018, 17:02:00
00000423.csv	1.5 kB	21.04.2018, 17:12:00
00000435.csv	1.1 kB	21.04.2018, 17:24:00
00000448.csv	1.4 kB	21.04.2018, 17:38:00
00000450.csv	1.1 kB	21.04.2018, 17:40:00
00000451.csv	1.1 kB	21.04.2018, 17:40:00
00000462.csv	1.1 kB	12.04.2018, 17:53:00
00000470.csv	394 B	21.04.2018, 18:00:00
00000472.csv	95 B	21.04.2018, 18:02:00
00000474.csv	394 B	21.04.2018, 18:03:00
00000478.csv	72 B	21.04.2018, 18:06:00
00000479.csv	279 B	21.04.2018, 18:06:00
00000480.csv	302 B	21.04.2018, 18:07:00
00000481.csv	302 B	21.04.2018, 18:07:00
00000482.csv	279 B	21.04.2018, 18:07:00
00000483.csv	279 B	21.04.2018, 18:07:00
00000488.csv	187 B	21.04.2018, 18:11:00
00000489.csv	187 B	21.04.2018, 18:12:00
00000490.csv	72 B	21.04.2018, 18:12:00
00000492.csv	164 B	21.04.2018, 18:14:00
00000518.csv	1.3 kB	21.04.2018, 18:42:00
00000539.csv	1.1 kB	21.04.2018, 19:03:00



### Рисунок 3.3. Файлы на ftp-серверах

Сначала происходит подключение к ftp-серверу, затем вводится указанный логин и пароль, переходим по прописанному заранее пути к специальной папке, после чего получаем список файлов в ней. Затем осуществляется проверка, если есть хоть один csv файл, то скачиваем его и помещаем в папку files, после успешного скачивания помещаем его в массив. После чего закрываем соединение с сервером, после скачивания необходимых данных, он нам больше не нужен.

Данные в файлах представлены в виде строки (рис.3.4.), нам нужно их корректно занести в свою базу данных. Мы перебираем в цикле файлы, скачанные с сервера, достаём данные csv файла, если файл не пустой, то извлекаем из него информацию, разбивая строку. Записываем в базу данных тип счётчика (m1), далее из строк отделяем дату от времени, конвертируем дату в формат ГГГГ-ММ-ДД и записываем в переменную, далее записываем время и последняя строка несёт информацию о показаниях счётчика. Название файла является номером счётчика, записываем всю информацию о нём в базу, если за эту дату данных еще нет. После проделанных действий с файлом, мы его удаляем, так как он нам больше не нужен.

```
//dd.MM.yyyy hh:mm:ss;m1
13.04.2018 06:00:53;1
13.04.2018 15:59:13;1
14.04.2018 03:40:09;1
14.04.2018 15:57:13;1
15.04.2018 03:47:05;1
15.04.2018 16:12:56;1
16.04.2018 03:37:15;1
16.04.2018 15:51:02;1
17.04.2018 03:36:11;2
17.04.2018 15:45:37;2
18.04.2018 03:36:18;2
18.04.2018 15:38:08;2
19.04.2018 03:34:28;2
19.04.2018 16:29:10;2
20.04.2018 03:35:06;2
20.04.2018 15:45:26;2
21.04.2018 03:36:49;2
21.04.2018 15:46:34;3
```

### Рисунок 3.4. Пример данных на ftp-серверах

Весь наш веб-сайт представляет собой иерархию папок, каждая из которых хранит специальные скрипты для функциональности системы.

cron/ftpdata.php - скрипт, который запускается раз в сутки, скачивает csv файлы с удалённого сервера и записывает показания из этих файлов в нашу базу.

css/ - папка в которой находятся файлы стилей (для оформления сайта).

files/ - папка в которую закачиваются скачанные csv файлы.

img/ - папка с картинками, которые используются на сайте.

includes/ - папка, в которой находятся небольшие html шаблоны, которые нужно подключать в разных частях страницы.

js/ - папка в которой находятся js файлы.

lib/ - папка с библиотеками.

plugins/ - здесь размещены сторонние скрипты (скрипт календаря).

modules/ - здесь все модули сайта (учётные записи, тарифы, пользователи, дома, личный кабинет).

config.php - пароли для подключения к базе данных и к удалённому серверу по ftp.

index.php - главный скрипт, через который осуществляется вся работа.

tmp.inc.php - главный html шаблон сайта.

В папке modules:

- /название\_модуля/css/style.css - оформление модуля

- /название\_модуля/includes/ - html шаблоны, которые подключаются в разных частях html шаблона модуля (всплывающие формы, например, форма для добавления пользователя)

- /название\_модуля/tmp.inc.php - html шаблон модуля.

- /название\_модуля/index.php - php код модуля, который нужен для вывода информации при открытии страницы данного модуля.

- /название\_модуля/ajax.php - код, который обрабатывает асинхронные запросы (все действия, которые совершаются без перезагрузки страницы, например, добавить или удалить пользователя).

Интерфейс информационной системы мониторинга и анализа данных о продажах питьевой воды

При открытии веб-сайта пользователь видит перед собой форму для входа (рис. 3.5) куда ему следует ввести логин и пароль, которые он получает от вышестоящего управления – администратор от разработчика системы, управляющий от администратора, собственник квартиры от управляющего или администратора. При неверном вводе данных пользователь увидит перед собой форму с предупреждением – «Неправильный логин или пароль».



Е-mail

Пароль

Войти



### Рисунок 3.5. Вход в систему

Полный доступ к системе имеет только администратор, пользователи с другими ролями имеют некоторые ограничения, поэтому мы рассмотрим функциональность системы через роль администратора.

При входе в систему вниманию предоставляется главная форма (рис.3.6), где пользователь видит в верхней части экрана меню вкладок, а именно: «учётные записи», «тарифы», «пользователи», «дома», «выйти». В правом верхнем углу имя и фамилия администратора с возможностью редактирования своего профиля при переходе в настройки.

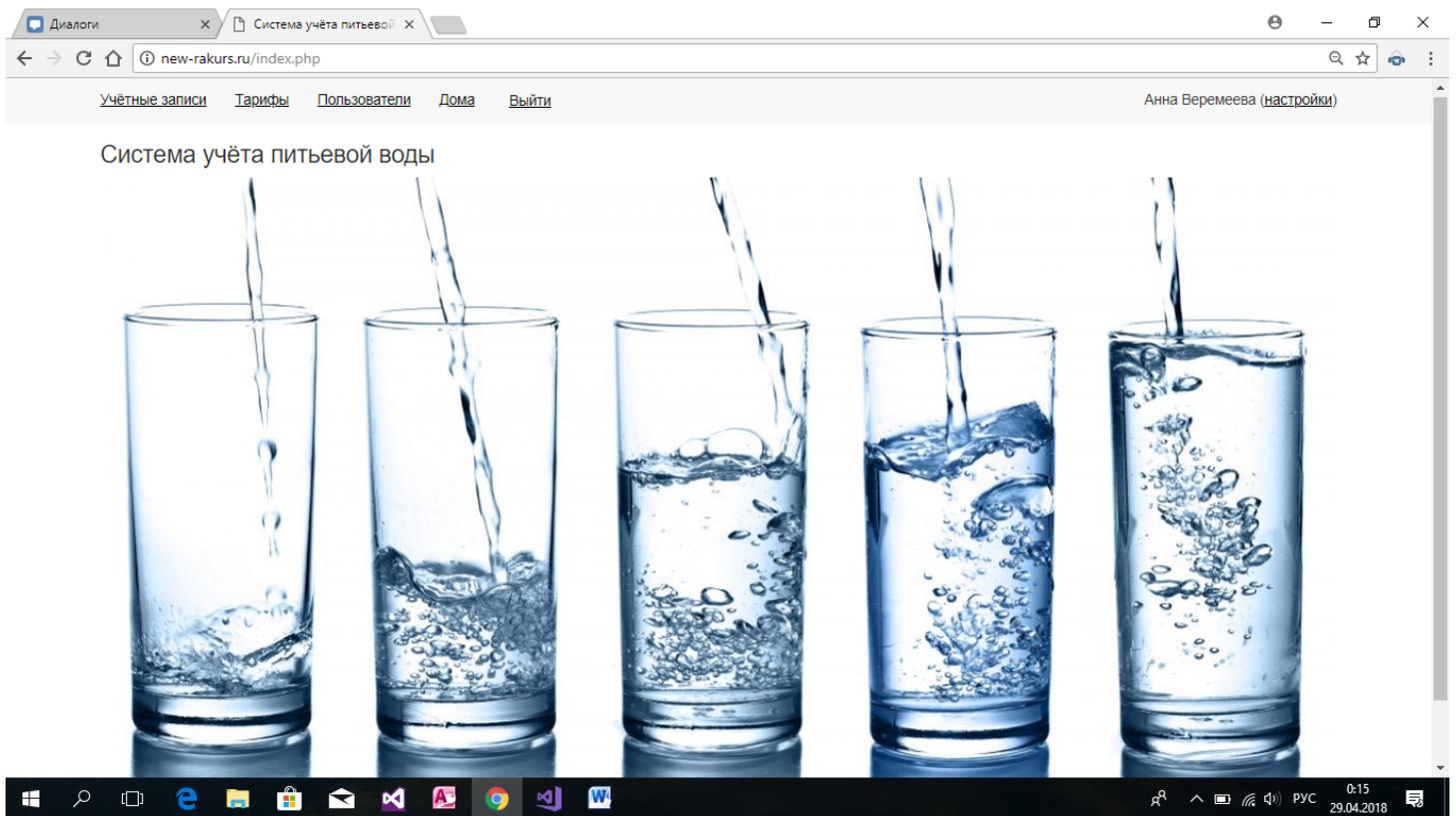


Рисунок 3.6. Главная форма

Если администратор хочет отредактировать свои данные, то он переходит в настройки, расположенные рядом с его именем и фамилией. Открывается специальная форма (рис. 3.7) для редактирования, после чего изменения сохраняются при нажатии соответствующей кнопки.

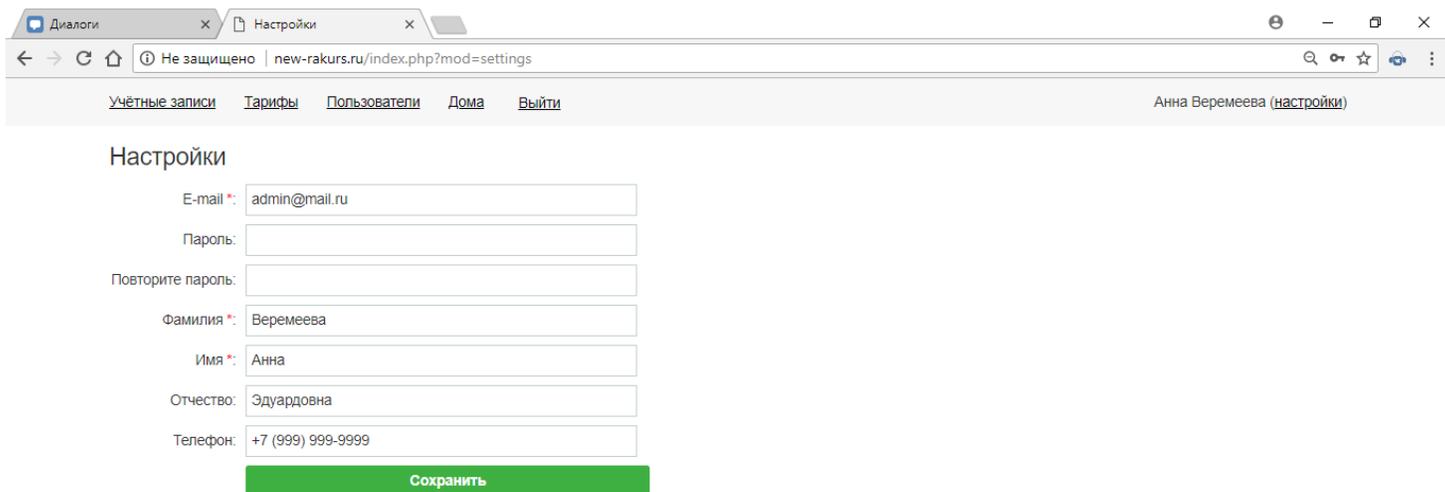


Рисунок 3.7. Настройки профиля

Первая вкладка на главной форме «учётные записи» (рис.3.8), в ней хранится информация об управляющих. Можно получить информацию о фамилии, имени и отчестве и e-mail. Администратор вправе редактировать информацию об управляющем, а также он может удалить желаемого пользователя из базы, для этого нужно нажать соответствующие кнопки.

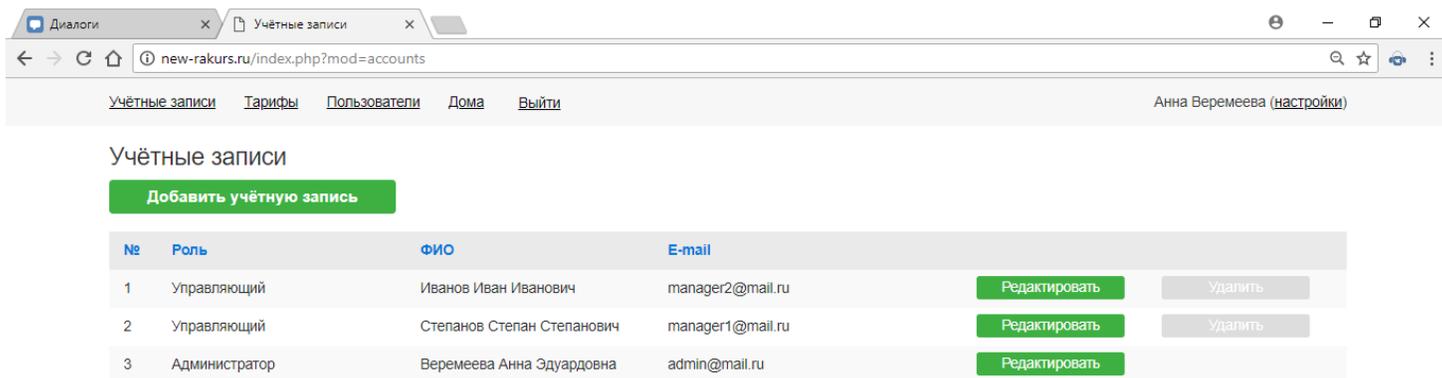


Рисунок 3.8. Учётные записи

На форме имеется специальная кнопка «добавить учётную запись», при её нажатии на экране появятся специальные поля для заполнения (рис. 3.9). Требуется указать следующую информацию: роль нового пользователя (администратор/управляющий), указать его e-mail, который в дальнейшем будет служить логином, пароль, фамилию, имя, отчество и контактный телефон.

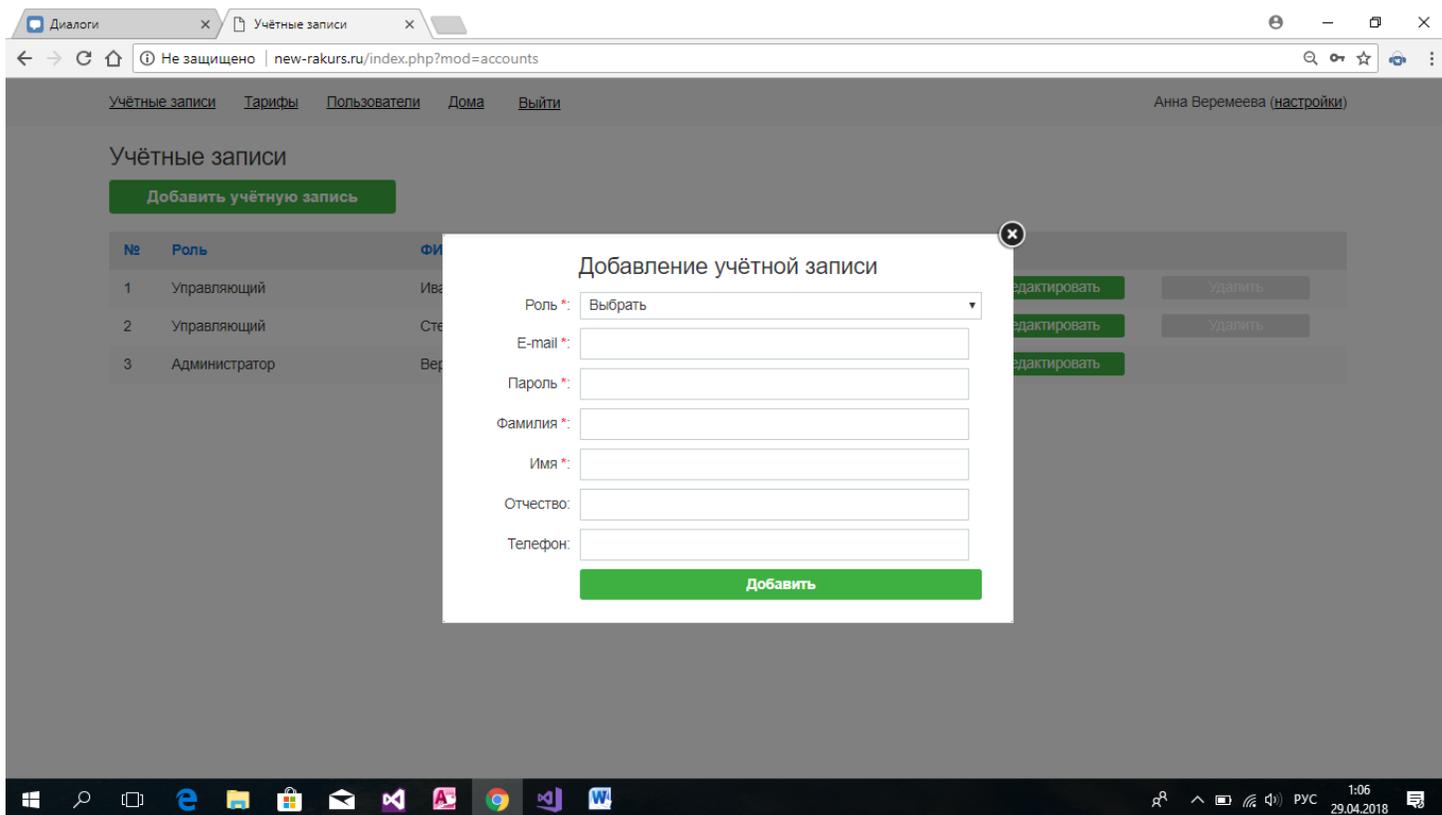


Рисунок 3.9. Добавление учётной записи

Вкладка «тарифы» несёт информацию об условиях потребления (рис.3.10). Так при переходе на этот раздел, на форме открывается информация о названии тарифа, его типе, единицы измерения и стоимость. Администратору предоставляется возможность добавить новый тариф, заполнив соответствующие поля, а также можно отредактировать существующий или удалить из базы.

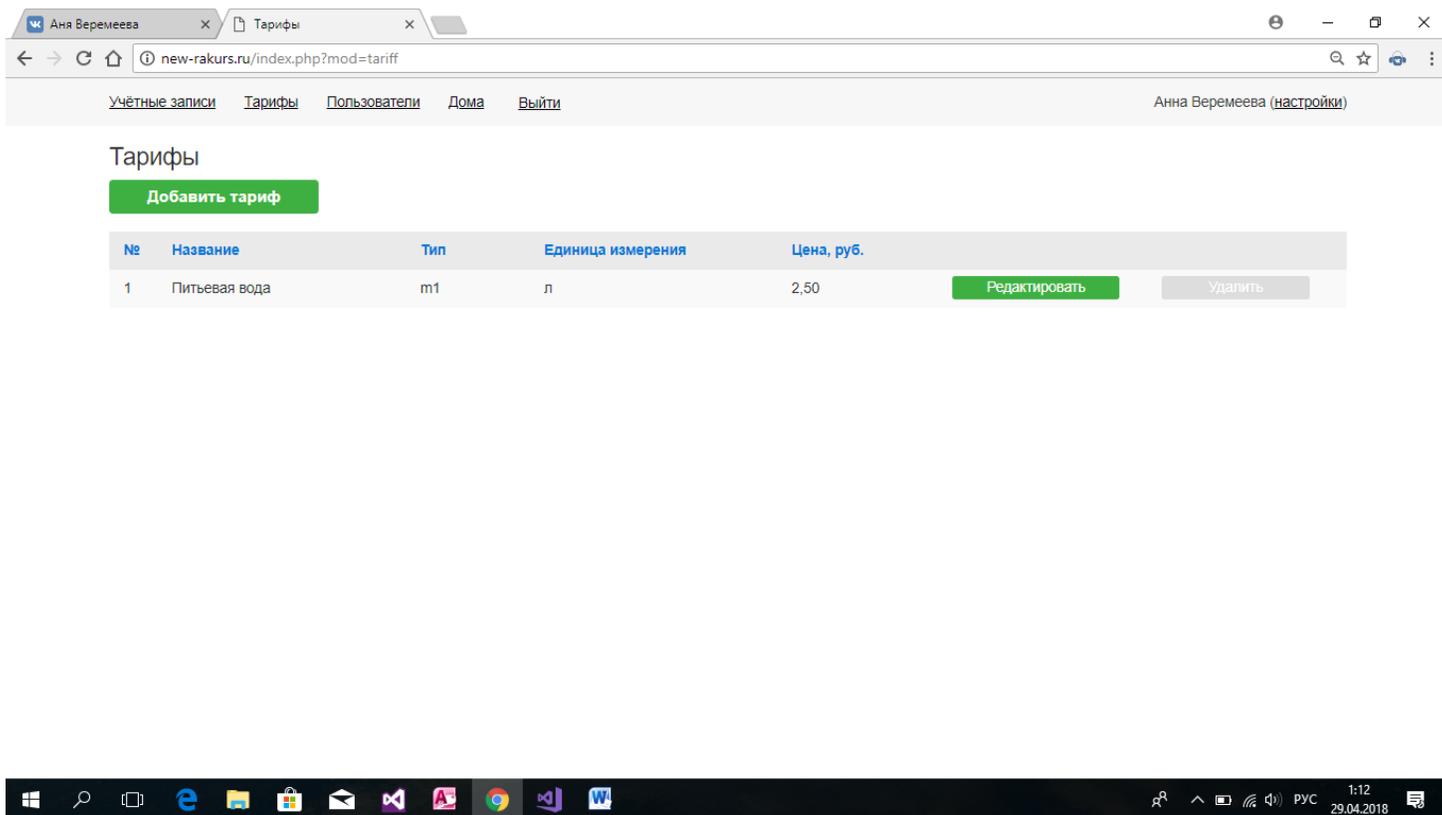


Рисунок 3.10. Тарифы

Вкладка «пользователи» содержит информацию обо всех собственниках квартир, которые зарегистрированы в системе (рис. 3.11). На форме можно увидеть номер договора с клиентом, его фамилию, имя и отчество, адрес проживания, номер квартиры, можно посмотреть статистику потребления воды, нажав на адрес, можно привязать еще одну квартиру к пользователю. Также, можно просмотреть количество квартир, закрепленных за пользователем, а при необходимости добавить новые. Аналогично, как и в учётных записях, пользователей можно редактировать, удалять и соответственно создавать новых (при заключении договора с клиентом).

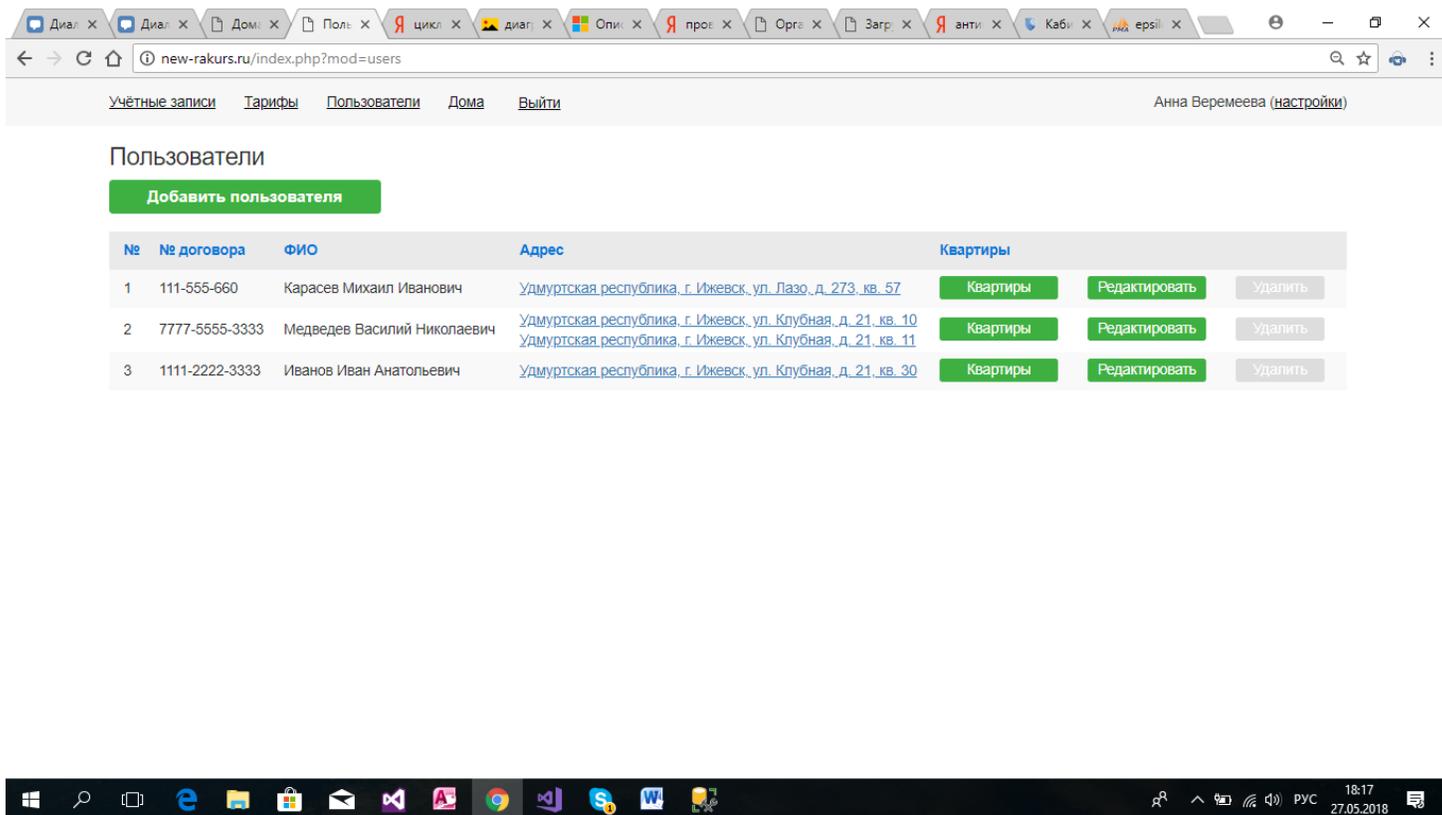


Рисунок 3.11. Пользователи

При нажатии на статистику пользователя, открывается новая страница с анализом данных потребления питьевой воды (рис.3.12). Можно ввести интересующий промежуток времени и нажать «показать», после чего появятся данные по используемому тарифу, об общем количестве расходуемых литров и общую сумму. Ниже будет представлена таблица, где можно увидеть в какой день какие были данные. После таблицы статистика будет представлена в графическом виде для лучшего понимания (рис. 3.13).

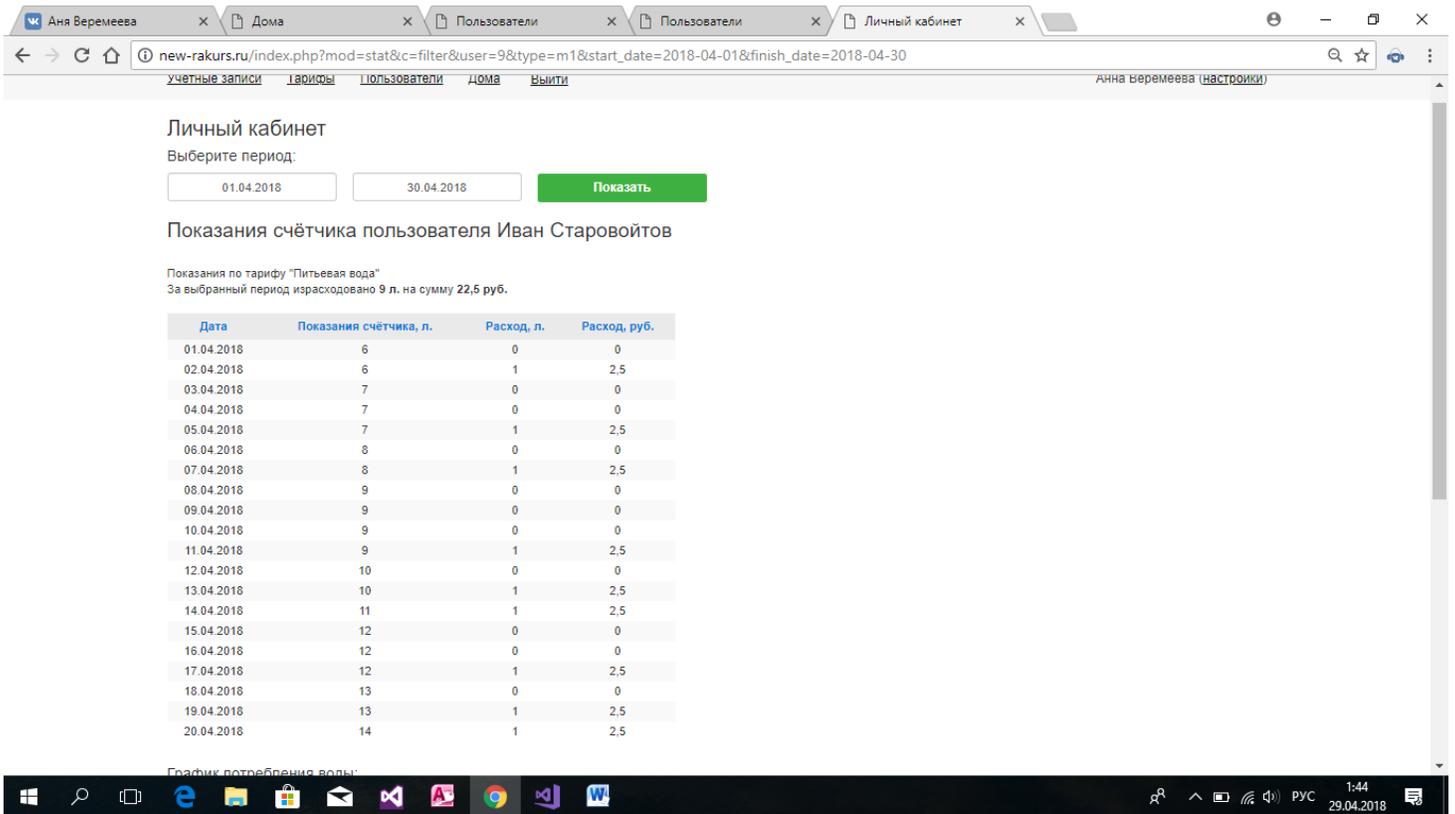


Рисунок 3.12. Статистика потребления пользователя

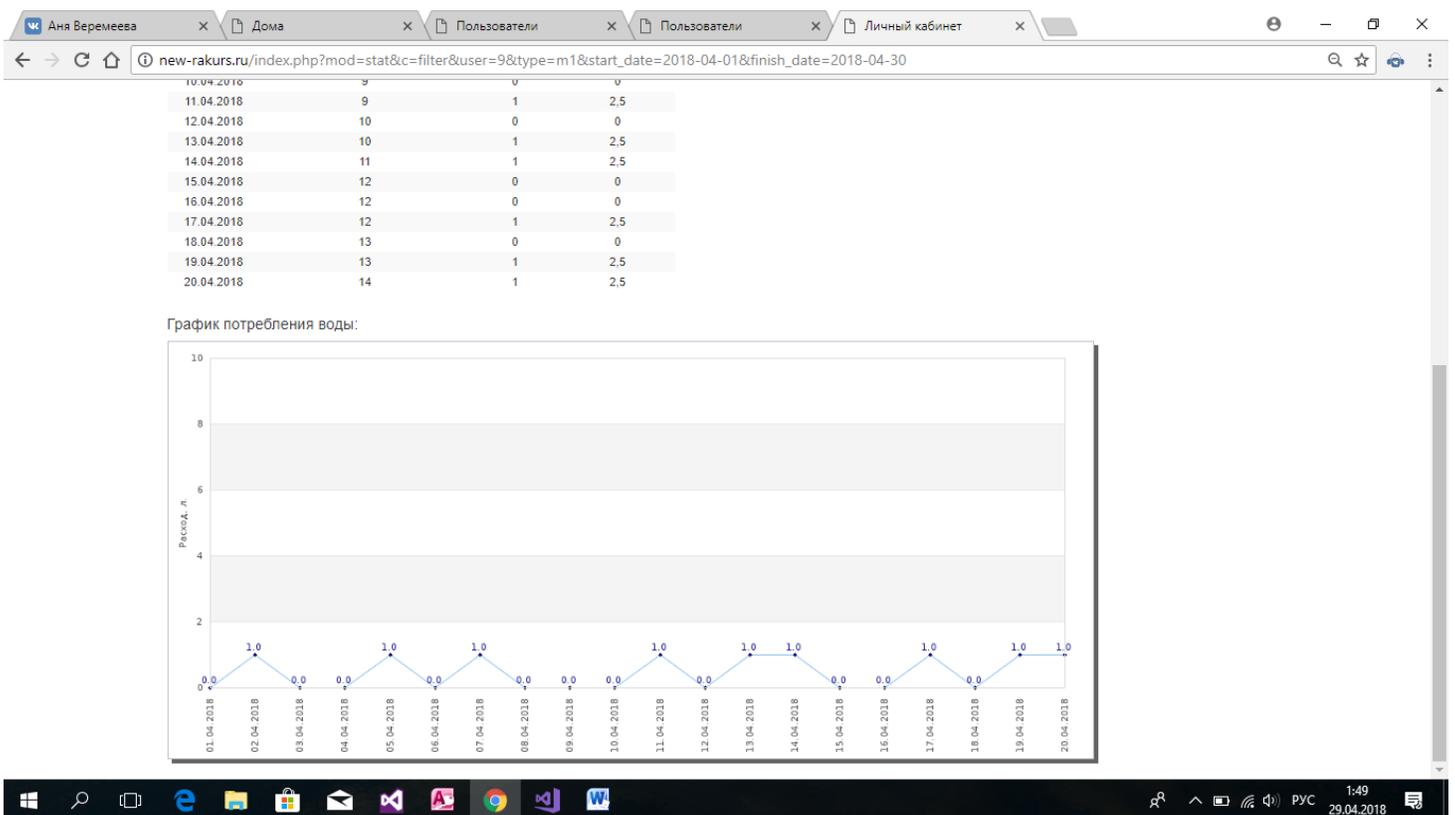


Рисунок 3.13. График потребления воды

Вкладка «дома» предоставляет администратору информацию о зданиях, которые подключены к системе (рис.3.14). На форме выведен адрес дома, информация о потребителях и мониторинг данных. Есть возможность: добавления нового дома в систему, при заполнении специальных полей и сохранении в базу, редактирования и удаления.

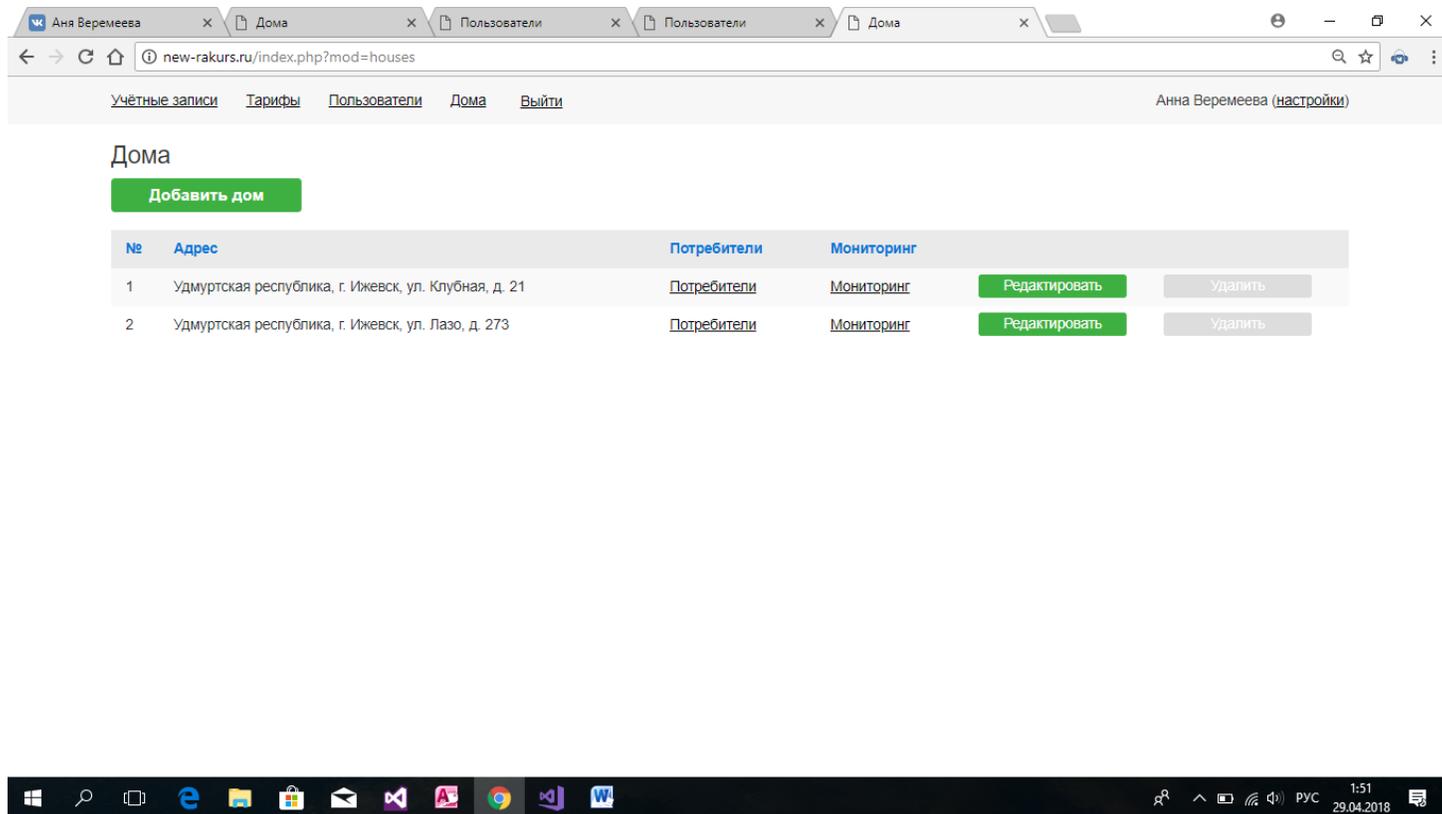


Рисунок 3.14. Дома

Если в разделе «дома» нажать на поле «потребители», то система выведет на экран всех клиентов этого дома (рис.3.15). После чего можно посмотреть статистику по каждому из них.

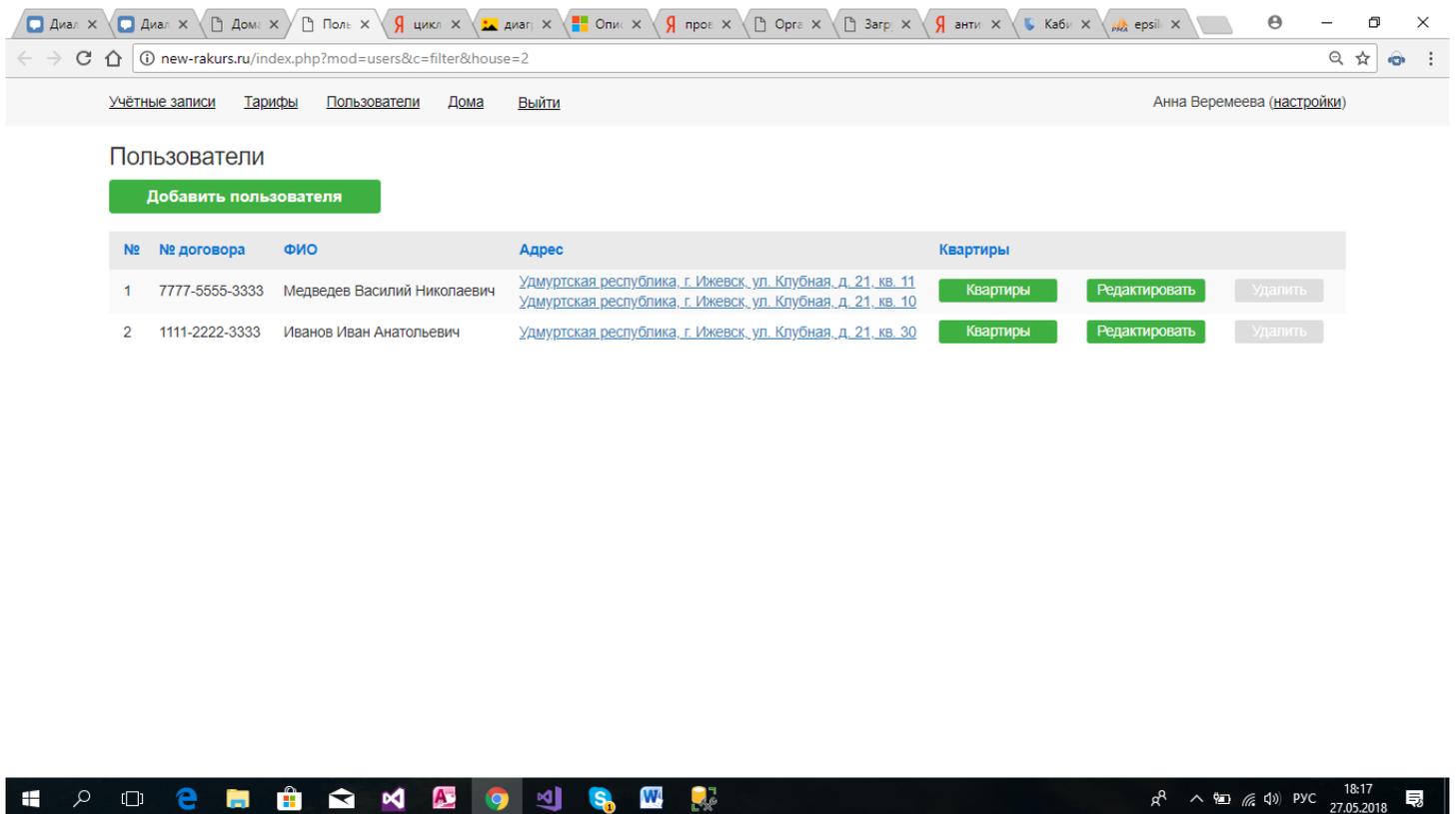


Рисунок 3.15. Список пользователей определенного дома

Если в разделе «дома» нажать на поле «мониторинг», то на форме появится общая информация потребления питьевой воды по всему дому, аналогично, как и у пользователей в виде таблицы и графика.

Тестирование информационной системы мониторинга и анализа данных о продажах питьевой воды

Для того чтобы программа работала корректно, её следует протестировать на все возможные входные данные и проверить, выдаёт ли программа верные результаты и не заканчивает ли свою работу раньше времени.

Тестирование будем выполнять последовательно по имеющимся вкладкам в начальной форме и по кнопкам в них. Осуществлять проверку по ролям будем от администратора до собственника квартиры.

Вся информация должна быть проверена на её корректность. Должна быть осуществлена проверка на вход по логину и паролю, должны открываться вкладки для администратора: учётные записи, тарифы, пользователи, дома, выйти, настройки пользователя. При неверном указании логина или пароля должно выводиться предупреждающее сообщение. Во вкладке учётные записи должна

быть информация об управляющих и администраторах, возможность добавления нового пользователя, редактирование имеющейся информации, удаление любой строки. При незаполненных полях после проведения манипуляций с данными, должно выводиться предупреждающее сообщение.

Вкладка тарифы должна выдавать корректную информацию о тарифах, должна быть осуществлена проверка на возможность изменять, удалять и добавлять новые данные.

Раздел пользователи должен корректно выдавать информацию о собственниках квартиры, требуется проверить добавление, удаление, редактирование пользователей. Напротив каждого пользователя должна быть возможность просматривать статистику потребления. Требуется осуществить проверку на корректность, сравнивая данные с формы с данными на ftp-серверах. Статистика должна выводиться за вводимый администратором период в виде таблицы, где по каждому дню будут предоставлены данные по количеству потребления, также должно быть протестировано корректное выведение на графике. Необходимо осуществить проверку на правильность расчета общей суммы потребления по указанному тарифу. Проверить вывод общей информации по количеству употребленных литров и затраченной суммой за определенный период.

Раздел дома должен, верно, выдавать информацию о домах, которые занесены в базу данных. Должно быть корректное добавление, редактирование, удаление данных.

В процессе тестирования нужно осуществить проверку на вход, что логин и пароль собственника открывает форму только по статистике данных в его квартире. Логин и пароль управляющего предоставляет доступ к данным по управляемому дому. Логин и пароль администратора предоставляет полный доступ к системе.

Более того, требуется проверить каждую форму на правильность ввода данных, так, в поле, где требуется ввести телефон, должен отображаться шаблон +7 (...) ..-...-..., куда пользователь имеет право ввести только цифры, не должно быть допуска к написанию букв. Аналогично, со всеми полями, куда требуется ввести только цифры, не должно быть никаких буквенных вводов.

Должна быть осуществлена проверка на достаточность вводимой информации, так, если в поля под красной звездочкой (обязательные к заполнению) оставить пустыми, должно быть выведено на экран соответствующее сообщение.

Проверив все вышеизложенное, можно утверждать, что веб-сайт работает корректно и стабильно.

Полное тестирование приведено в Приложении В.

## **ЗАКЛЮЧЕНИЕ**

Все чаще люди перестают использовать воду из-под крана, в качестве питьевой. Исходя из этой проблемы, были созданы специальные службы, чтобы проводить отдельный источник качественной и безопасной для жизни воды. К сожалению, существующие на сегодняшний момент решения по контролю потребления питьевой воды, имеет некоторые недостатки. Именно в этом и выразилась актуальность нашей работы – в создании удобной системы для мониторинга и анализа данных о продажах питьевой воды, как для управляющего персонала, так и для потребителей.

В процессе работы мы решили следующие задачи:

1. Проанализировали сферу продаж:
  1. Изучили методы анализа продаж, которые подошли к нашей работе, используя имеющуюся литературу в данной тематике.
  2. Изучили актуальные способы мониторинга продаж, проанализировав современные пособия в данной сфере.
2. Провели обзор существующих решений, сравнив их преимущества и недостатки в общей таблице.
3. Разработали архитектуру информационной системы и алгоритм работы сервера.
4. Выполнили последовательно все этапы разработки программного продукта.

Все поставленные ранее задачи были решены, можно сказать, что главная цель, которая заключалась в проектировании и разработке информационной системы мониторинга и анализа данных о продажах питьевой воды достигнута.

Таким образом, после завершения этапа разработки, можно начинать подготовку к этапу внедрения, так как разработанная информационная система может использоваться для анализа и мониторинга данных о продажах питьевой воды в многоквартирных домах и на предприятиях с любого устройства и локации, что будет полезно многим сегодня.

# СПИСОК ЛИТЕРАТУРЫ

1. Как проводить анализ продаж: этапы, методы и способы. URL: <http://kakzarabativat.ru/marketing/analiz-prodazh/> (Дата обращения: 13.11.2019).
2. Шнаппауф Р. Практика продаж/ Р.Шнаппауф. -М.: Интерэксперт, 2007. 448 с.
3. Анализ продаж. URL: <http://worldsellers.ru/analiz-prodazh/> (Дата обращения: 13.11.2019).
4. Edwards, R. D., & Magee J. (2001). Technical analysis of stock trends. San Francisco: Golden Gate University
5. Weetman, P. (2006) Management Accounting: Second edition. England: Pearson Education Limited
6. АВС АНАЛИЗ ПРОДАЖ. ПРИМЕР РАСЧЕТА В EXCEL. URL: <http://finzz.ru/abc-analiz-prodazh-primer-v-excel.html> (Дата обращения: 13.11.2019).
7. Справочник по html. URL: <http://htmlbook.ru/HTML> (дата обращения: 13.11.2019).
8. Meloni, J., & Telles, M. (2008). PHP 6 FAST & EASY WEB DEVELOPMENT. Boston: Course Technology
9. Скотт, Б. Проектирование веб-интерфейсов/ Б.Скотт, Т.Нейл. - Символ-Плюс 2010