НАУЧНАЯ РАБОТА

Название работы: **«АВТОМАТИЗИРОВАННАЯ ОНЛАЙН СИСТЕМА УЧЕТА УСПЕВАЕМОСТИ ОБУЧАЮЩИХСЯ ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЙ РЕСПУБЛИКИ КРЫМ»**

Авторы работы: Осыка Виктория Евгеньевна, Горщар Роман Сергеевич

Учебное заведение (научно-исследовательское учреждение):

Гуманитарно-педагогическая академия (филиал) ФГАОУ ВО «Крымский федеральный университет имени В. И. Вернадского» в г. Ялте

Факультет: Институт экономики и управления, направление подготовки 09.04.03 «Прикладная информатика», курс: 2 магистратуры

**Научный руководитель**: **Филимоненкова Татьяна Николаевна, старший преподаватель**

**Содержание**

[ВВЕДЕНИЕ 3](#_Toc494142632)

[РАЗДЕЛ 1. АНАЛИЗ ДЕЯТЕЛЬНОСТИ ДИРЕКЦИИ ИНСТИТУТА ПО УЧЕТУ УСПЕВАЕМОСТИ ОБУЧАЮЩИХСЯ 4](#_Toc494142633)

[РАЗДЕЛ 2. АНАЛИЗ И ОБОСНОВАНИЕ ВЫБОРА ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ СОЗДАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ 12](#_Toc494142634)

[РАЗДЕЛ 3. РАЗРАБОТКА И ТЕСТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ 16](#_Toc494142635)

[ЗАКЛЮЧЕНИЕ 32](#_Toc494142636)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 34](#_Toc494142637)

# **ВВЕДЕНИЕ**

В настоящее время в условиях стремительного развития информационных систем и технологий автоматизация деятельности высших учебных заведений Республики Крым приобретает особую актуальность. Использование современных компьютерных технологий позволяет упростить работу сотрудникам, оптимизировать процесс управления и обеспечить конкурентоспособность вуза. Работа с большим количеством информации об успеваемости обучающихся в вузе является одной из важнейших задач, подлежащих автоматизации.

Один из важнейших аспектов деятельности ВУЗа является учет успеваемости обучающихся. С каждым днем в учебном заведении увеличивается поток информации, которую становиться все сложнее обрабатывать вручную. К тому же такой труд можно назвать неэффективным, так как сейчас есть возможность доверить эту работу офисной технике, которая выполнит все качественно и кратчайшие сроки. Создание информационной системы учета сведений успеваемости обучающихся позволит сократить объем рутинной работы, требующей скрупулёзной точности и оперативности обработки данных.

Цель данного проекта – разработка информационной системы учета промежуточной и итоговой аттестации обучающихся высших учебных заведений Республики Крым.

Объект исследования – документооборот в дирекции образовательного учреждения Республики Крым.

Предмет исследования – автоматизированное решение задач, таких как обработка и хранение информации о промежуточной и итоговой аттестации (электронных ведомостей и журналов), формирование статистических отчетов по группам и факультетам, выписка листов пересдач и контроль успеваемости студентов.

# **РАЗДЕЛ 1. АНАЛИЗ ДЕЯТЕЛЬНОСТИ ДИРЕКЦИИ ИНСТИТУТА ПО УЧЕТУ УСПЕВАЕМОСТИ ОБУЧАЮЩИХСЯ**

Дирекция института является тем подразделением, которое осуществляет административные функции и обеспечивает работу с обучающимися ВУЗа. На Дирекцию возлагаются следующие обязанности:

* организация воспитательной и учебно-методической работы в институте;
* планирование организационно-методической, научно-методической, учебно-методической деятельности и формирование отчетности по ним;
* организация и выполнение образовательно-профессиональных программ по специальностям и направлениям института;
* повышение профессионального уровня сотрудников института;
* определение перспективных направлений научно-исследовательской деятельности института, помощь кафедрам института в их реализации;
* обеспечение документооборота и делопроизводства по функционированию института.

Учет успеваемости, включающий оценку и контроль навыков и умений обучающихся, качеств, которые характеризуют основы их профессионального уровня, дает необходимый воспитательный и учебный эффект только в том случае, когда он проводится систематически и своевременно, а не от случая к случаю. Оперативная оценка и контроль учебных успехов обучающихся, являются ключевыми факторами на пути к улучшению качества образовательного процесса.

Учет может подразделяться на периодический, текущий и итоговый. Текущий учет успеваемости по теоретическим предметам предполагает систематическую проверку преподавателем имеющихся знаний у обучающихся по изученному учебному материалу непосредственно на занятиях. Выставленную оценку педагог доводит до сведения учащегося и, если возникает необходимость, указывает ему на пробелы в его знаниях или имеющиеся недостатки, а также предлагает пути их исправления. Оценки текущего контроля в пятибалльной системе заносятся преподавателем в журналы академических групп, на специально отведенных для каждой дисциплины страницах. Периодический учет успеваемости осуществляется с целью определения степени усвоения обучающимися изученного материала и качество умений, знаний и навыков, приобретаемых ими в течение семестра.

Суть периодического учета успеваемости – это определение качества полученных умений, навыков и знаний по темам программы, пройденным в текущем учебном полугодии. Он заключается в обязательном выполнении каждым обучающимся проверочной работы. Итоги работ выставляются преподавателем в специальную ведомость, а затем в зачетную книжку студента.

Итоговый учет успеваемости должен определять прочность и полноту приобретенных обучающимися знаний и умений за семестр в соответствии с требованиями учебных программ. По некоторым дисциплинам итоговые оценки могут выставляться преподавателем на основании текущих отметок, получаемых обучающимся в течение семестра. Если же студент не согласен с таким результатом, ему будет предложено сдать зачет. По остальным дисциплинам, установленным учебным планом, итоговый учетосуществляется с помощью экзаменов. Экзамены и зачеты проводятся строго по расписанию, которое составляет методист дирекции. В расписании указывается группа, дата, время, аудитория, дисциплина, преподаватель и данные о пересдачах. Расписание утверждается директором института и своевременно доводится до сведения преподавателей и обучающихся.

Экзамены и зачеты принимает преподаватель, проводивший учебные занятия. На экзамене допускается присутствие куратора группы, руководителей и других сотрудников института. Преподаватель выставляет оценку обучающемуся за его ответы по билету на основании специальных критериев оценок успеваемости. После завершения экзамена преподаватель заполняет сформированную сотрудниками дирекции экзаменационную ведомость. Готовая подписанная ведомость хранится в архиве дирекции.

Обучающимся, которые не явились на экзамен по неуважительной причине, и тем, кто получил на экзамене оценку «неудовлетворительно», дается шанс пересдать экзамен в назначенное согласно расписанию время. В этом случае сотрудником дирекции формируется специальный бланк, который в дальнейшем прикрепляется к основной экзаменационной ведомости.

На основании результатов сессии решается вопрос о переводе обучающихся на следующий курс.

Исходя из анализа задач учета сведений об успеваемости обучающихся, была построена схема информационных потоков (рис.1.1).

**Преподаватель**

**ИС Учета промежуточной и итоговой аттестации обучающихся**

Выставленная оценка

Данные о выставленных ранее оценках

**Дирекция ВУЗа**

Данные об обучающихся

Данные о группах и учебном плане на текущий семестр

Запросы аналитики и отчеты об успеваемости обучающихся,

и групп

Данные о кафедрах, специальностях и профилях подготовки

Отчеты и анализ успеваемости обучающихся и групп

Рис.1.1. Диаграмма информационных потоков системы учета сведений об успеваемости обучающихся

Схема позволяет специфицировать как обрабатываемые информационной системой данные, так и ее функции. Система, представленная в виде иерархии потоков информации, которые описывают асинхронный процесс реорганизации данных с момента первичного ввода в систему до выдачи конечного результата пользователю.

В процессе проектирования значительное внимание уделяется детальной разработке архитектуры информационной системы в целом, а также проектных решений по отдельным ее объектам и элементам, их анализу, практической апробации и внедрению [4].

Инфологическое (концептуальное) проектирование базы данных (БД) подразумевает сбор и анализ требований к системе, описание структуры и динамики программного продукта, характера информационных потребностей пользователей в доступных и не зависимых от реализации базы данных терминах, а также создание информационной модели организации, полностью независящей от различных деталей реализации [1].

Дирекция института будет использовать систему для получения точной и оперативной отчетности об успеваемости обучающихся на протяжении семестра или года обучения. Для того, чтобы предоставлять требуемую отчетность, система обязана своевременно получать и хранить необходимое количество данных об успеваемости.

В течение всего периода обучения вводом данных в систему должны заниматься преподаватели, т. е. система напрямую участвует в процессе учета успеваемости, также она является инструментом управления и хранения журналов успеваемости академических групп. Подобный способ ввода данных предоставит возможность генерации качественных и оперативных отчетов об успеваемости на протяжении семестра или всего периода обучения и позволит избежать дублирования учетной работы. ~~В таком случае,~~ Для того, чтобы это обеспечить, в системе ~~необходимо~~ требуется хранить дополнительные данные, необходимые для учета успеваемости – списки институтов, кафедр, групп, дисциплин, специальностей, данные об обучающихся и преподавателях и т. п.

Стоит отметить, что большинство преподавателей не заинтересованы в использовании системы, так как она не предлагает им серьезных преимуществ по сравнению с существующими средствами учета успеваемости, но требует больших затратов сил и времени для перехода к ее использованию. Проблема усугубляется тем, что, некоторые преподаватели имеют небольшой или отсутствующий опыт работы с компьютером. Использование и освоение в повседневной работе новой информационной системы является для них сложной задачей. Поэтому возникает необходимость разработки максимально функционального интерфейса, но в тоже время простого и понятного в использовании.

Но с другой стороны, регулярный и своевременный ввод данных об успеваемости обучающихся, а также их полнота имеют огромное значение для продуктивного использования данной системы дирекцией института.

Успешное внедрение и эксплуатация информационной системы напрямую зависит от того, насколько просто будет организована работа в ней для преподавателей. Это означает, что система должна иметь «дружественный» интерфейс и требовать минимальных трудовых и временных затрат, но тем не менее обеспечивать решение задачи учета успеваемости.

С учетом вышесказанного была разработана инфологическая модель учета успеваемости обучающихся. Данная модель учитывает объединение учебных групп по кафедрам и тот факт, что педагоги могут совмещать работу на нескольких кафедрах.

На рис. 1.2. представлена концептуальная модель системы учета успеваемости обучающихся.

Данная модель отображает взаимосвязь между сущностями системы и может служить основанием для разработки физической модели конкретной базы данных.

Факультет

Кафедра

Группа

Преподаватель

Дисциплина

Учебный план

Студент

Состоит из

Закреплена за

Состоит из

Закреплена за

Закреплен за

Состоит из

Закреплен за

Состоит из

Закреплена за

ведет

Посещает

Состоит из

Входит в

Журнал учета работы групп

Ведет

Ведомость

Данные входят в

Заполняет

Рис.1.2. Концептуальная модель предметной области

Для разработки физической модели базы данных учета успеваемости обучающихся ВУЗа необходимо проанализировать предметную область, взяв за основу первичные документы учета успеваемости обучающихся.

В данном случае – это журнал и ведомость учета успеваемости академических групп. Журнал ведется преподавателями ВУЗа. В нем учитываются все проводимые занятия, посещаемость, самостоятельная работа обучающихся, а также выставляются оценки текущего контроля, оценки за экзамены и другие виды аттестаций, выводится итоговая оценка за семестр. Во время занятий журналы должны находиться у преподавателей.

Экзаменационную ведомость оформляет сотрудник дирекции института. В ней содержатся данные о проводимом экзамене или зачете, а именно: наименование структурного подразделения, институт, направление подготовки, курс, группа, дисциплина, преподаватель, ФИО обучающегося, № зачетной книжки, дата. Далее ведомость регистрируется в журнале выдачи с присвоенным ей уникального регистрационного номера, а затем передаётся преподавателю, осуществляющему прием экзамена или зачета по данной дисциплине. В свою очередь, преподаватель выставляет оценки в трех форматах: по национальной шкале, по 100 бальной шкале и ЕСТS.

На рис. 1.3. приведена схема базы данных учета успеваемости. Рассмотрим ее подробнее.

Одной из основных сущностей, информация о которой должна храниться в базе данных, является «Ведомость учета успеваемости». В качестве первичного ключа (*англ*. Primary Key - PK) используется уникальный номер ведомости.

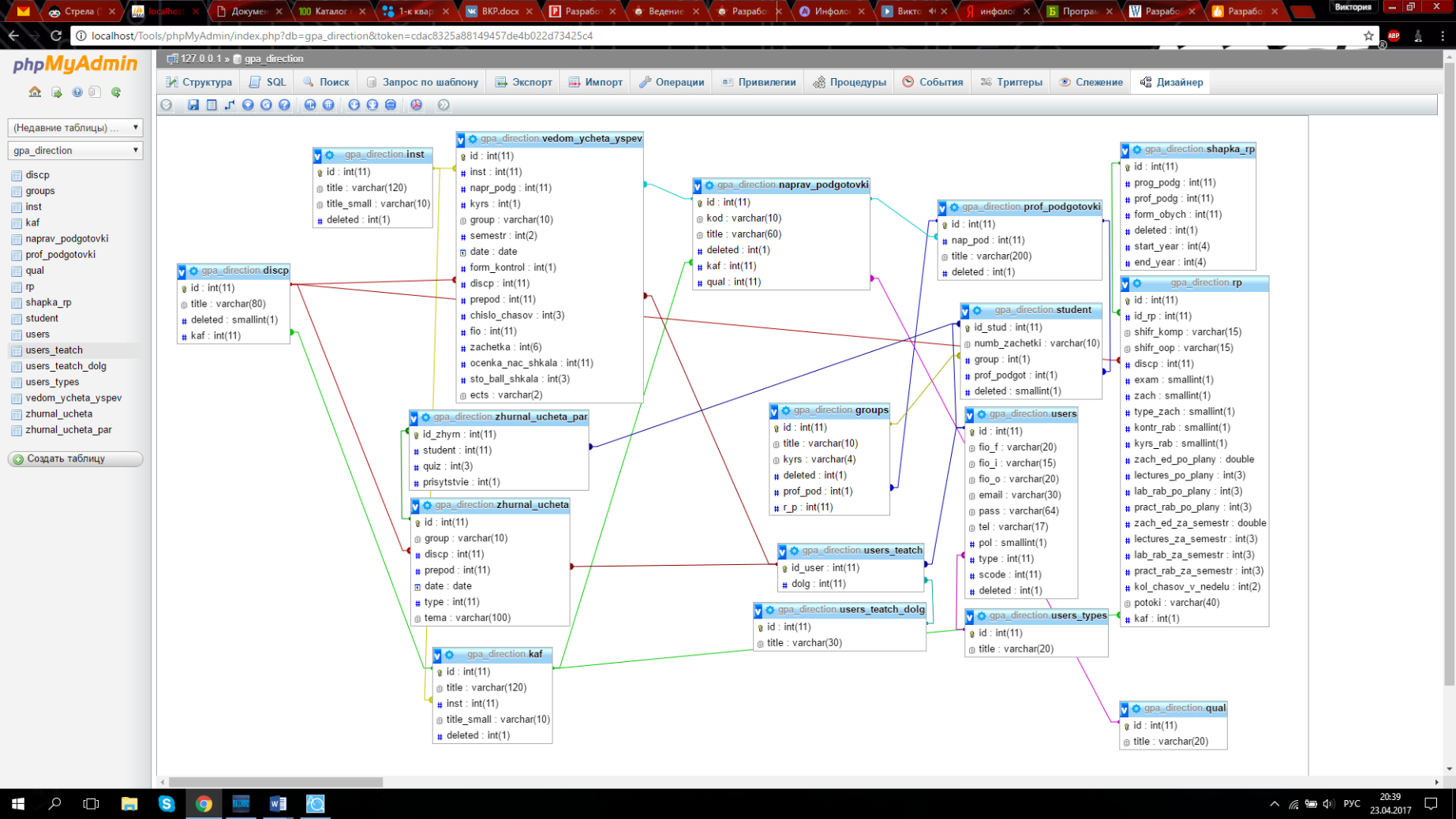


Рис.1.3. Физическая модель базы данных учета успеваемости обучающихся

Поскольку предполагается, что информационная система будет многопользовательской с разграничением прав доступа и системой паролей, особое внимание необходимо уделить сущности «Пользователи», первичный ключ которой – номер пользователя. Было принято решение соединить все категории пользователей в одну таблицу, чтобы в дальнейшем избежать проблем с идентификацией в системе. Так как удаление какого-либо пользователя может привести к некорректному отображению определенных данных или сбою в выполнении запросов, было принято решение не удалять информацию, а присваивать ей определенный статус. Для этого предназначен атрибут deleted (статус удаления). По умолчанию значение данного атрибута – «0». Значение «1» ставится в случае, если администратор решит удалить данные.

Вспомогательные сущности: «Дисциплины», «Институты», «Кафедры», «Направления подготовки», «Профиль подготовки», «Тип пользователей», «Должности» – содержат справочную информацию и связываются с другими сущностями внешними ключами.

Если проанализировать данную модель, можно сделать вывод, что модель данных полностью удовлетворяет требованиям третьей нормальной формы (3НФ), так как устранены атрибуты, зависящие от атрибутов, не входящих в уникальный идентификатор. Такие атрибуты являются основой отдельной сущности.

*( Везде нужно убрать будущую форму – система разработана, внедрена – имеется акт о внедрении, наоборот писать как свершившийся . Я исправила пару предложений ниже, дальше посмотри и исправь сама.)*

~~Планируется, что~~ Информационная система ~~будет~~ реализована как веб-приложение с возможностью многопользовательского доступа по сети. Такой подход к архитектуре приложения ~~позволит~~ позволил обеспечить доступ к базе данных преподавателям, которые будут вносить сведения об итогах промежуточной аттестации непосредственно в базу данных, тем самым исключив из процесса внесения данных промежуточное звено в виде работника дирекции института, что в свою очередь повысит достоверность информации.

~~Таким образом, можно сделать вывод, что~~ Разработанная модель данных легла в основу создания автоматизированной, многопользовательской системы, ~~которая позволит~~ позволяющей своевременно, качественно и надежно обрабатывать большие объемы информации учета успеваемости обучающихся ВУЗа, ~~Также появится возможность~~ предотвратить различные ошибки, совершаемые сотрудниками ВУЗа, или подделку данных.

# **РАЗДЕЛ 2. АНАЛИЗ И ОБОСНОВАНИЕ ВЫБОРА ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ СОЗДАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ**

Современные средства разработки можно разделить на несколько категорий. Первая – CASE-средства (независимые (upper CASE), и интегрированные с СУБД), которые позволяют проектировать БД и приложения в сочетании с встроенными средствами разработки «клиент-серверных» приложений (Designer/2000+Developer/2000).

Вторая категория состоит из отдельных средств разработки и проектирования информационных систем, реализующих различные методологии, в комплексе со средствами создания приложений. К подобной категории можно отнести такие средства, как ERwin/ERX+PowerBuilder, SILVERRUN+JAM и др [8].

До недавнего времени для автоматизации разработки и проектирования информационных систем применялась структурная методология, подразумевающая использование формализованных методов описания разрабатываемой системы и принимаемых технических решений. Также использовались графические средства описания различных моделей информационных систем с помощью диаграмм и схем, что приводило к значительным трудозатратам. Подобные обстоятельства в значительной степени повлияли на появление программных средств, которые получили название Computer Aided Software Engineering (CASE-средства). Термин CASE означает разработку программного обеспечения средствами компьютера.

CASE-технологии представляют собой методы проектирования информационных систем, а также набор инструментальных средств, позволяющих в наглядной форме моделировать предметную область, анализировать эту модель на всех этапах разработки и сопровождения системы и разрабатывать приложения в соответствии с информационными потребностями пользователей [17].

CASE-средства позволяют целиком разрабатывать информационные системы (логику процессов, функциональные спецификации, интерфейс для пользователей и базы данных), обеспечивая при этом правильный процесс их разработки. Инструменты таких категорий обладают определенными сложностями, высокой гибкостью и широкой областью применения.

Представление о функциональных возможностях развитых средств разработки может дать краткая сравнительная характеристика программ, приведенная в табл. 2.1.

Таблица 2.1

Характеристики средств разработки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **West-mount I-CASE + Uniface** | **Designer/2000+ Developer/2000** | **SILVER-RUN + JAM** | **ERwin/ERX + Builder** |
| **Поддержка полного жизненного цикла ИС** | + | + | + | + |
| **Обеспечение целостности проекта** | + | + | - | - |
| **Независимость от платформы** | + (ORACLE, Informix, Sybase, Ingres и др.) | - (целевая СУБД - только ORACLE) | + (ORACLE, Informix, Sybase и др.) | + (ORACLE, Informix, Sybase, ODBC) |
| **Одновременная групповая разработка БД и приложений** | + | - | - | - |

Анализ приведенных данных, указывает на то, что из перечисленных программ лишь комплекс Westmount I-CASE+Uniface наиболее удовлетворяет критериям, взятым в качестве главных. Следует отметить, что каждая из программ является одной из наиболее мощных в своей категории.

Применение определенного средства не исключает возможности использования в том же проекте других средств.

В процессе выбора интерфейса программного комплекса следует сразу же отметить, что текстовые и речевые интерфейсы не рассматриваются, так как с их помощью не получится организовать удобную, функциональную информационную систему. С учетом требований к разрабатываемой системе больше всего подходит графический интерфейс, который в свою очередь разделяется на оконный и веб-ориентированный.

Со временем разработка и тестирование оконных приложений усложнились, появилось множество асинхронных событий, серверных, таймерных, сигналов от других программ и окон. Среда информационной системы стала менее стабильной и предсказуемой. Возможности для разработки оконных приложений увеличились, но формирование архитектур и подходов запоздало. Естественно крупные специализированные фирмы и опытные разработчики создают качественные программы, но остальная часть как пользователей, так и разработчиков – пользуются «любительскими» интерфейсами.

В отличии от оконных приложений, веб-интерфейс не требует инсталляции клиентов на компьютеры пользователей. Не нужно беспокоиться по поводу версий .NET, версий DLL и большого количества настроек, решать конфликты программного обеспечения на пользовательских компьютерах. Все операции с данными происходят в браузерах, которые в данный момент почти все одинаково отображают информацию. Обновления софта не требуется. Все данные размещены на серверах, где есть возможность осуществления резервного копирования и защиты.

Таким образом можно сделать вывод, что для информационных систем больше всего подходит веб-интерфейс. Веб приложения не требуют установки на компьютер пользователя. Для обеспечения полноценной работы необходим доступ в интернет и браузер. Не требуется специальная настройка и администрирование веб-интерфейса, администраторами системы являются ее разработчики. Для нормальной работы интерфейса требуется минимальная конфигурация компьютера. Обновление происходит автоматически. Важно, что система обладает высокой мобильностью – возможностью использования разработки в любом месте, где доступен интернет.

Правильный подход к выбору средств разработки – важнейшая задача организации комплекса работ по созданию информационной системы. Проанализировав технологии и программные решения, изучив их недостатки и достоинства, был осуществлен выбор средств разработки.

В качестве среды разработки выбран Adobe Dreamweaver СС, приложение, которое предназначено для создания профессиональных веб- ориентированных систем. Программа поддерживает широкий спектр современных технологий: HTML 5, ASP.NET, AJAX, XML и классические CSS и JavaScript [12]. Имеются три режима создания и редактирования страницы: Код, Разделение, Дизайн. Среда также работает со страницами, написанными на скриптовом языке PHP. Поэтому можно создавать не только статические страницы, но и динамические веб-ориентированные системы, с подключением баз данных.

Инсталляцию и конфигурирование компонентов можно осуществить автоматически, с помощью Денвера. Стандартный пакет включает в себя:

* Установщик.
* HTTP-сервер Apache – кроссплатформенное ПО, позволяющее использовать СУБД для аутентификации пользователей, подключать внешние модули для предоставления данных, модифицировать сообщения об ошибках и т. д.
* PHP5 с поддержкой MySQL.
* Система управления: запуск, перезагрузка и завершение всех компонентов Денвера.
* phpMyAdmin - это веб-приложение, представляющее собой интерфейс для администрирования СУБД MySQL. Оно позволяет запускать команды SQL и просматривать структуру базы или ее содержимое. ПО имеет свободную лицензию, и постоянно модернизируется с учетом быстрого развития технологий.

# **РАЗДЕЛ 3. РАЗРАБОТКА И ТЕСТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ**

Качественная разработка базы данных способна обеспечить простоту ее поддержки в дальнейшем. Данные должны храниться в таблицах, с учетом, что каждая таблица содержит информацию одного типа, например, сведения о кафедрах. Тогда если возникнет необходимость обновления конкретной информации, будет достаточно изменить ее в одном месте, чтобы обновленные данные отображалась во всех остальных таблицах базы.

При проектировании базы данных были учтены функции, которые должны выполняться информационной системой учета успеваемости обучающихся. Процедура разработки приложения включает в себя создание 18 таблиц, таких как: «Дисциплины», «Группы», «Институты», «Кафедры», «Направления подготовки», «Профиль подготовки», «Рабочие планы», «Студенты», «Преподаватели», «Журнал», «Ведомость», а также другие вспомогательные таблицы, и 25 подчиненных отношений, которые обеспечивают корректность хранения данных. Все названия таблиц формировались по принципу классификации хранимой в них информации, что позволит упростить процесс работы с базой данных, а также создание запросов к ней. С этой же целью к каждому атрибуту прописаны специальные комментарии, которые содержат необходимую для разработчика информацию, в частности варианты возможных значений. Для управления данными необходимо создание соответствующего интерфейса приложения и написание программного кода, обеспечивающего необходимый функционал.

После создания всех таблиц, определяются первичные ключи, а также с помощью внешних ключей устанавливаются связи между таблицами, что обеспечивает целостность базы данных. Большинство связей в базе данных имеют тип «один ко многим».

После создания всех таблиц, полей и связей была произведена проверка структуры базы данных и выявлены возможные недочеты. Это необходимо делать именно на данном этапе, пока таблицы не заполнены данными. Чтобы проверить правильность связей в базе данных, были выполнены тестовые запросы для получения нужных сведений, а также созданы черновые формы и отчеты для отображения ожидаемых данных. Найдены и исключены все излишние повторы информации.

Используемый в разработанной базе данный подход к хранению информации предусматривает необходимость перехода данных в стадию синтаксического анализа с последующим представлением в виде, необходимом для работы алгоритмов. Применение такого метода хранения данных позволяет существенно уменьшить нагрузку при выполнении запросов, сократить количество записей, а соответственно и пространство в базе данных.

Интерфейс имеет важное значение для информационной системы и является главной ее составляющей, прежде всего направленной на конечного пользователя. Именно по интерфейсу зачастую пользователи судят о программе в целом. Так же часто программу оценивают по степени комфортности и понятности интерфейса. ~~Вместе с тем, проектирование и разработка интерфейса довольно трудоемкое дело~~ [13].

Веб-интерфейс, который был создан ~~создается~~ для информационной системы учета успеваемости обучающихся – это рабочий инструмент, а не произведение искусства. Поэтому его нужно сделать как можно более удобным для разных типов пользователей. Для этого все страницы системы необходимо оформить в одном стиле.

Так как проектирование осуществлялась ранее, то на основе разработанных моделей системы было осуществлено прототипирование в электронной форме следующих видов страниц: главной страницы, панели редактирования данных (рис.3.1), страниц, имеющих табличное отображение информации. Все они состоят из четырех главных блоков: контактная панель, меню, блок контента и футер сайта. Этих схем необходимо придерживаться во время разработки веб-ресурса.

Главная страница системы не перегружена информацией, что является немаловажным для быстрой адаптации новых пользователей. Так как сайт не всегда посещают люди, имеющие аккаунт в системе, было решено разместить контактную панель вверху страницы. Воспользовавшись указанными данными, человек может получить информацию нужным ему способом и в кратчайший срок. Логотип и название веб-ресурса размещены по стандарту – в верхней части страницы. Там же размещается и меню. Главная страница в первую очередь предназначена для идентификации пользователя в системе. По центру экрана размещена форма входа, куда нужно вводить e-mail и пароль. Эти данные можно получить у администратора системы.

Контактная панель

Логотип

Название сайта

Главное меню

Дополнительная информация

Тип данных (например, дисциплины)

Тип данных

Тип данных

Тип данных

Тип данных

Данные

Форма ввода и редактирования

Данные

Кнопки управления

Кнопки управления

**Управление** (добавление, изменение, удаление)

Личный кабинет

Заголовок страницы

Рис.3.1. Схема интерфейса панели редактирования данных

Макет страниц, изображенный на рис.3.1, доступен только пользователям с правами администратор или сотрудник ВУЗа. Страница разработана в минималистическом стиле. Отсутствуют отвлекающие элементы в виде изображений, баннеров, ярких кнопок и т.д. Имеющаяся в базе информация будет отображается в виде списка, её можно редактировать и удалять, с помощью небольших кнопок управления.

Страницы, имеющие табличное отображение данных, должны иметь методы, которые позволяют делать выборку из таблиц, а также добавлять и редактировать записи в них. Исходя из этого, в страницу включены следующие элементы: режим создания, редактирования и просмотра таблиц.

Следующим важным этапом в разработке макета сайта является создание навигационной схемы. Именно продуманная навигация позволит пользователям свободно ориентироваться в системе. Добавление иерархической навигации на все страницы в одно и тоже место означает, что пользователь способен перейти в любое место из текущего раздела. Цвета гиперссылок должны отличаться от обычного текста, ссылки, на которые наведен курсор, подсвечиваться, и уже посещенные ссылки окрашиваться в другой цвет. Кнопка текущей страницы в навигационной панели должна быть недоступна для клика [15].

Данная информационная система закрытого типа, то есть большинство функций доступно только зарегистрированным пользователям. Исходя из этого, обычному посетителю веб-ресурса открыта только главная страница. Для пользователей, имеющих аккаунт в системе, доступны разные виды функций, в зависимости от типа пользователя. На рис.3.2. отображены схематично главные пути, которые обеспечат посетителю комфортное перемещение по системе.

Важным атрибутом является тип пользователя, который определяет доступность функций в системе. Информация в личном кабинете разбита на категории, названия блоков в меню навигации соответствуют своему контенту.

Для обмена информацией с пользователем в данной системе предназначены формы ввода. В формах можно размещать особые элементы управления, автоматизирующие ввод. Все преимущества использования форм открываются тогда, когда данные вводятся с заполненных бланков. В таком случае формы делаются с помощью графических средств так, чтобы они напоминали или повторяли оформление бланка.

Главная страница

Личный кабинет

Контакты

Рейтинг

Статистика

Местоположение

Преподаватель

Ведомости

Журнал групп

Студент

Успеваемость

Оповещения

Администратор

Регистрация пользователей

Панель редактирования

Сотрудник ВУЗа

Ведомости

Журнал групп

Панель редактирования

Отчетность

Рис.3.2. Краткая схема навигации по веб-ресурсу

Наиболее сложным элементом управления является таблица, поскольку она одновременно осуществляет ввод и вывод информации. Для каждой таблицы реализован свой собственный класс модели, описывающий информацию, отображаемую в данный момент времени, и логику поведения. Собственные модели также реализованы для определенных выпадающих списков, содержимое которых способно меняться динамически.

Проведенный анализ предметной области, разработанные логические и физические модели, а также спроектированный макет веб-интерфейса дают основание приступать к программной реализации информационной системы.

В первую очередь устанавливаются выбранные средства разработки, а именно Денвер. Далее необходимо позаботиться о правильности задания путей к файлам. Своевременно продуманная система именования файлов и каталогов на локальном компьютере, впоследствии сократит количество проблем при размещении веб-ресурса в сети Интернет. Важно учитывать возможность обновления системы в будущем. Чтобы при замене файлов, не менять все ссылки (так как это занимает много времени), рекомендуется использовать каталоги по типам файлов, не учитывая, к какой функциональной части сайта они относятся. В главном каталоге размещаются все страницы веб-ресурса. Прочие папки будут содержать изображения сайтов, описания внешнего вида документа и т.д. Каталог файлов, разработанный для данной информационной системы представлен на рис.3.3.

**WWW**

**Files**

**Страницы.php**

**CSS**

**IMG**

**FONT**

**PHP**

**JS**

**Файлы.css**

**Файлы.img**

**Шрифты**

**Файлы.php**

**Файлы.js**

Рис.3.3. Файловая структура системы

Основное преимущество такого подхода заключается в том, что файлы при обновлении заменяются всего один раз. Например, если один и тот же стиль используется на нескольких страницах, достаточно будет изменить его в одном месте, в данном случае в папке «CSS». При этом все ссылки на страницах будут указывать на замененный файл.

После того как выбран подход к файловой организации, можно приступить к установке выбранного визуального редактора кода DreamWeaver, и начать реализацию пользовательского интерфейса.

Сперва был создан основной дизайн, которой будет применяться ко всем последующим страницам. При подборе цвета фона было учтено, как он будет гармонировать со всеми остальными элементами: текстом, гиперссылками и др. Обеспечен достаточный цветовой контраст и контраст по яркости между навигационными элементами на странице и фоном. Меню сайта находится в правой части экрана и подсвечивается при наведении на него курсором. Так как система имеет разветвленную структуру, то ее страницы необходимо оформить в рамках единого стиля: использовать одни и те же управляющие элементы, одинаковые интервалы и подобные шрифты [10].

Так как главная страница является пропускным пунктом в систему, то следующий этап реализации будет начинаться именно с нее. Готовый шаблон заполняем контентом и функциональностью согласно разработанному макету. Перед отправкой формы, при изменении или добавлении значений, в полях ввода E-mail и Password осуществляется онлайн-проверка корректности.

Чтобы сообщить пользователю о правильности или неправильности введенных данных, разработана специальная функция «Event\_Color()». Окраска элемента происходит по следующему алгоритму: правильно введенные данные – выкрашиваются в белый цвет, неправильно – в ярко голубой.

Самый важный аспект системы, использующей учетные записи – это то, как защищены пароли пользователей. Существует несколько различных подходов к проблеме о том, как правильно осуществлять хеширование пароля. Хеш-алгоритмы – это односторонние функции. Предназначены они для превращения любого количества информации в «дактилоскопический отпечаток» фиксированной длины. Процесс хэширования не может быть обратим. При подобных алгоритмах даже незначительное изменение входного параметра в корне меняет получаемый хеш-код. Таким образом, пароли в данной системе проходят через безопасный криптографический алгоритм хеширования - SHA-2. Самый обычный пароль, состоящий только из цифр при сохранении в базе данных выглядит примерно таким образом: «b3978ae4a919ed9469bc8f03f02c390838d6e291026e44a1c9bd9485fa11ee5e».

Для восстановления доступа необходимо отправить запрос администратору, указав требуемые персональные данные и каким-либо образом их подтвердить. Например, прикрепить фотографию лица рядом с паспортом. Если данные верны, администратор назначает новый пароль и сообщает его пользователю.

После того, как данные полей удовлетворят требованиям, пользователь может нажать на кнопку «Вход». В этом случае прочитанные из формы авторизации значения отправляются в функцию проверки их наличия в базе данных.

Необходимо подчеркнуть, что в данном веб-ресурсе активно используются Куки. Куки – небольшие фрагменты данных, которые отправляются веб-сервером и хранятся на компьютере пользователя. Каждый раз, когда браузер открывает страницу соответствующего сайта, он пересылает эти данные веб-серверу в составе HTTP-запроса.

После авторизации в системе определяется тип пользователя, согласно которому система решает, какие страницы можно показывать, а какие для него недоступны. Помимо входа, на главной странице пользователь может выбрать один из четырех информационных разделов. Раздел «Помощь» включает в себя ответы на типовые вопросы, которые могут возникнуть в ходе использования системы. В блоке «Связь с администратором» можно узнать его контактные данные или связаться напрямую, через веб-ресурс и т.д.

Главная страница информационной системы учета успеваемости обучающихся представлена на рис.3.4.



Рис.3.4. Внешний вид главной страницы системы

Как уже говорилось ранее, веб-ориентированная система учета успеваемости обучающихся является многопользовательской. Следовательно, она должна иметь очень развитую подсистему администрирования, способную организовать корректную работу всех остальных модулей системы. Для реализации панели администрирования, потребуется определить ряд таблиц, которые должны хранить всю необходимую информацию для веб- ориентированной системы. Как правило такие таблицы, заполняются одноразово, и необходимость их редактирования возникает довольно редко. Например, таблица «Направления подготовки» или «Дисциплины». Панель администрирования изображена на рис.3.5.

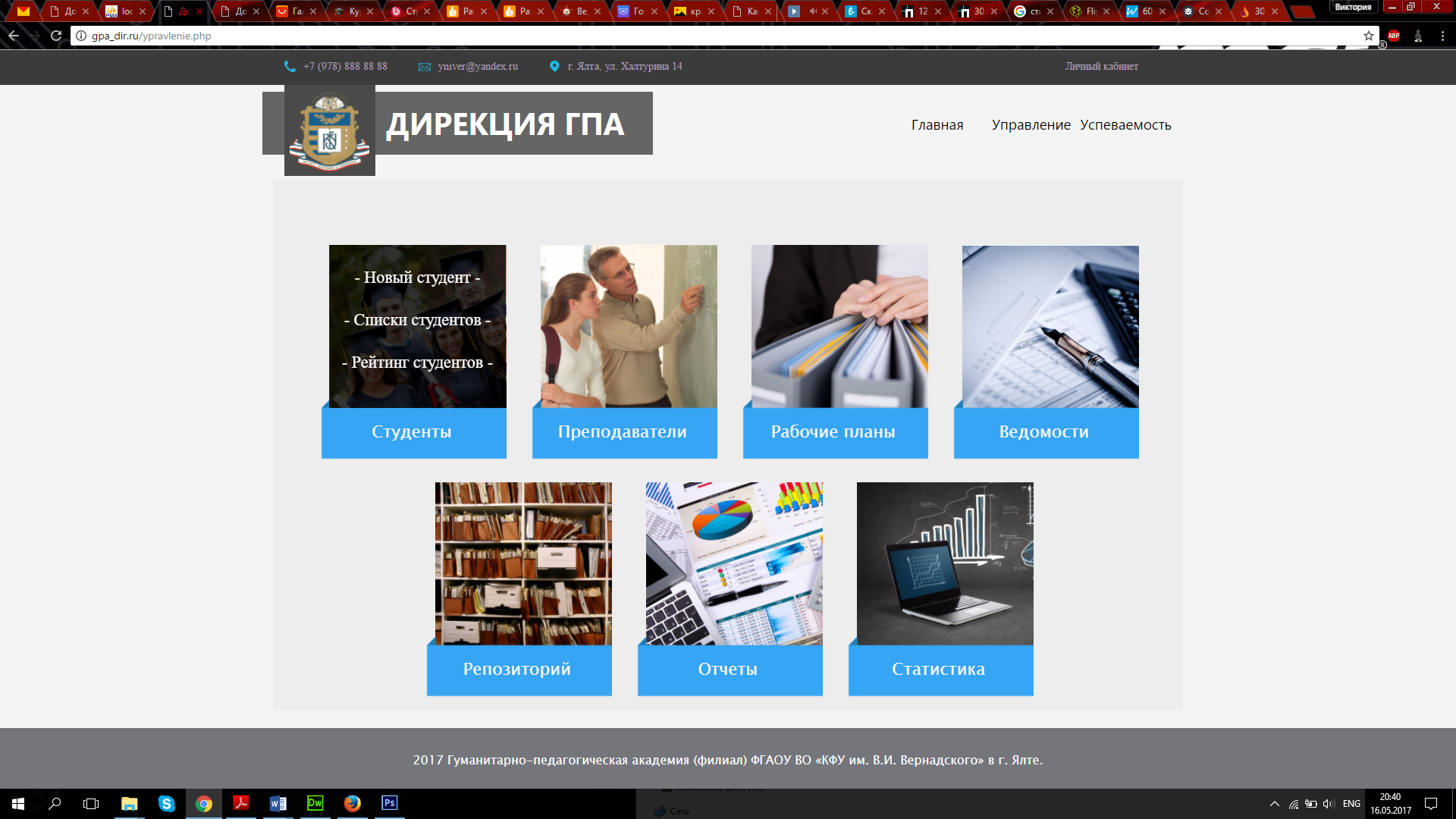


Рис.3.5. Панель управления для сотрудника дирекции

Для записи новой информации в таблицу предназначены специальные поля ввода и кнопки управления. При отправке данных на сервер, вначале проверяется их наличие в базе. Если совпадение не найдены, происходит запись. Такие проверки имеются не у всех полей, так как в некоторых случаях дублирование определенных записей допускается. Например, название направления подготовки «Прикладная информатика» может повторяться несколько раз, пока код направления будет отличным.

Если код и название направления совпадут с аналогичной записью в базе, система сообщит пользователю об ошибке посредством выполнения функции «Event\_Color()». Для изменения и удаления строки достаточно нажать на соответствующие кнопки. При всевозможных операциях с данными нумерация строк переопределяется в автоматическом режиме.

Следующей разрабатываемой страницей является регистрация пользователей с возможностью добавления, редактирования и удаления данных из системы. Изначально в простые поля ввода заносятся общие сведения о человеке, такие как фамилия, имя, номер телефона и т.д. В зависимости от выбранного типа пользователя подгружается дополнительная форма регистрации (рис.3.6). Например, для студентов необходимо также записывать номер зачетной книжки, направление подготовки, группу и тд. Вместе с формой ввода расширяется и фоновая картинка. После заполнения всех полей и успешного сохранения данных профиль пользователя можно будет отредактировать или удалить. Примерно по такому же принципу работает модуль формирования рабочего плана.

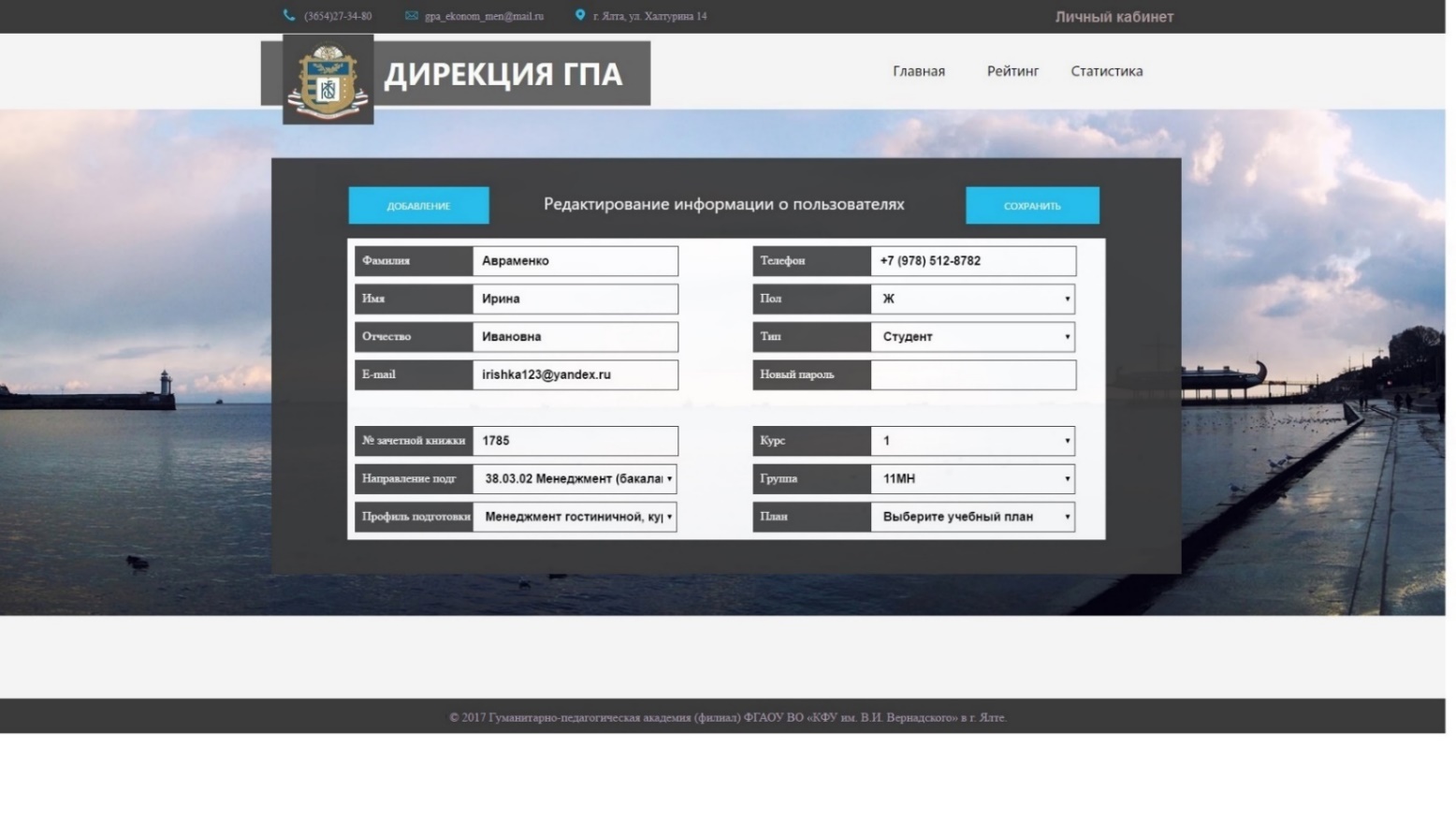


Рис.3.6. Страница редактирования пользователя типа «Студент»

При редактировании профиля загружаются все имеющиеся о нем данные из базы, кроме пароля. Если пользователю не требуется изменять пароль, это поле остается пустым. При отправке формы такой вариант учитывается и не считается за ошибку. В случае если профиля еще нет в базе данных (его только добавляют), пустое поле пароля приведет к ошибке. Если пароль утерян, он не может быть выслан пользователю на электронный ящик, так как администратор видит только зашифрованный пароль.

Страница «Рабочие планы» реализована согласно схеме интерфейса страниц с таблицами. Реализованы три варианта действий пользователя: создать, редактировать и просмотреть рабочий план. При создании потребуется заполнить основную информацию, такую как: направление подготовки, профиль подготовки, форму обучения, группу и др.

Вначале сотрудник дирекции может ввести данные только в первое поле. После выбора значения из выпадающего списка, открывается следующее поле с уже отсортированным массивом данных. Например, если выбрать программу подготовки «Бакалавр», в поле Направление подготовки будут отображаться не все возможные направления, а лишь те, которые удовлетворяют выбранному условию. Также динамически формируются данные в других полях текущей страницы.

Период начала и окончания обучения также проверяется в зависимости от выбранных ранее параметров. Например, если пользователь ввел год начала «2017», а год конца обучения «2022» для рабочего плана бакалавра, то система предупредит об ошибке, так как разница между годами не может быть больше четырех лет. После нажатия на кнопку «Сформировать» появится пустая таблица, которую можно продолжать заполнять.

После того, как пользователь успешно завершит заполнение таблицы, ему будет предложено просмотреть ее со всеми возможными подсчетами. Например, поле Общий объем при заполнении таблицы не отображается. Расчет выполняется программой на основании введенных ранее значений. В данном случае Общий объем = значения поля Зачетные единицы \* 36. Скрытые для редактирования поля можно увидеть на вкладке просмотра учебных планов (рис.3.7).

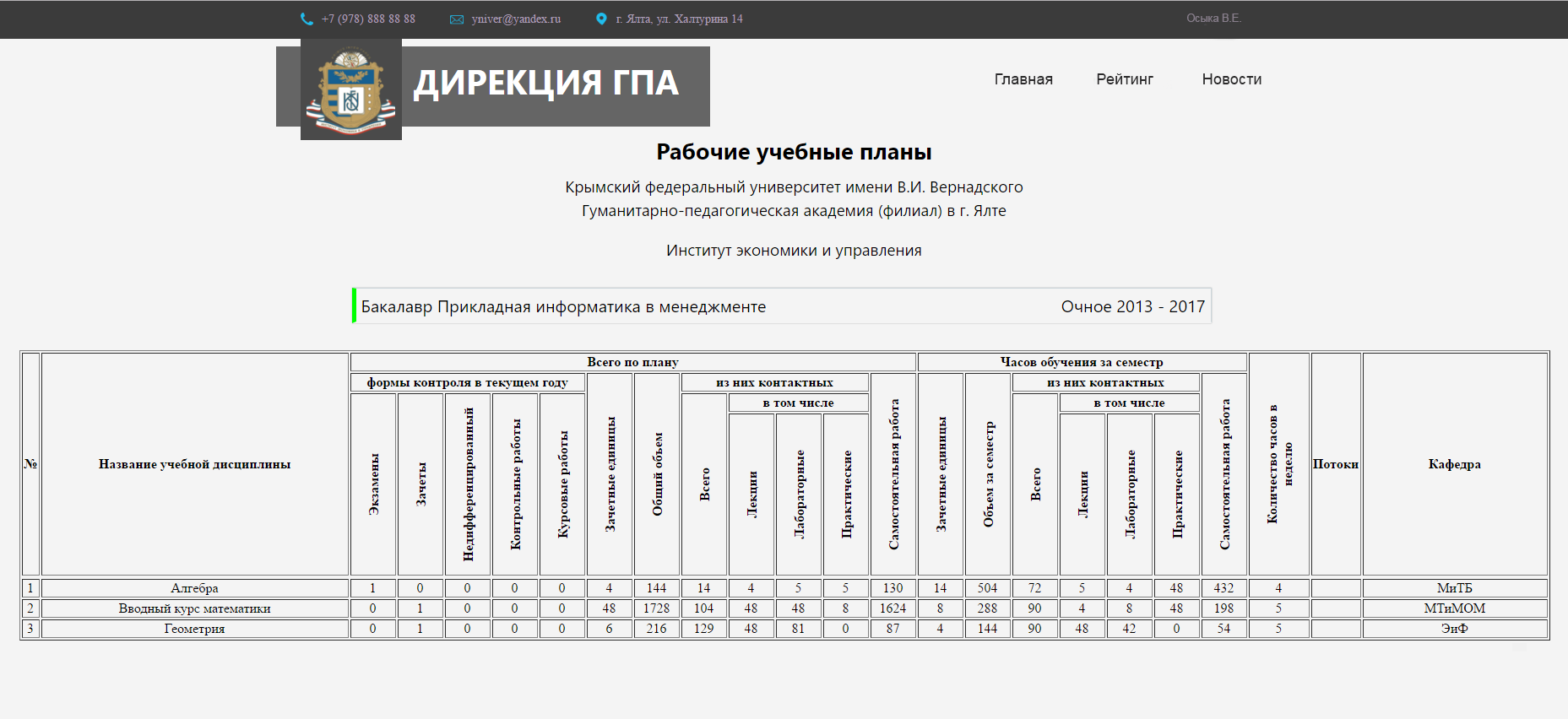


Рис.3.7. Страница просмотра учебного плана

Для информации, которая в данный момент неактуальна (например, данные о закончивших обучение) предназначены страницы архивов. Удобный поиск в таких архивах осуществляется благодаря «умным» фильтрам. «Умный» фильтр – функция отбора информации по заданным условиям, имеющая следующие особенности: показывает количество элементов, которые выберутся при использовании фильтра; в зависимости от выбранных ранее значений предлагает варианты комбинаций свойств; показывает пользователю то, что выбранные комбинации значений отсутствуют в базе данных.

В качестве примера архива можно привести страницу «Экзаменационные ведомости». Данная страница видна только сотруднику, который может выбрать и просмотреть действующую или закрытую ведомость. «Умный» фильтр отображает результат запроса на текущей странице веб-ресурса. В некоторых случаях поля ввода могут заблокироваться - это значит, что комбинация таких параметров невозможна, так как нет подходящих записей в базе данных.

После того, как пользователем произведен выбор, появится область с данными: ведомость и результаты сдачи экзамена (зачета). Когда все оценки выставлены, система посчитывает все необходимые значения и отображает результат (рис.3.8).

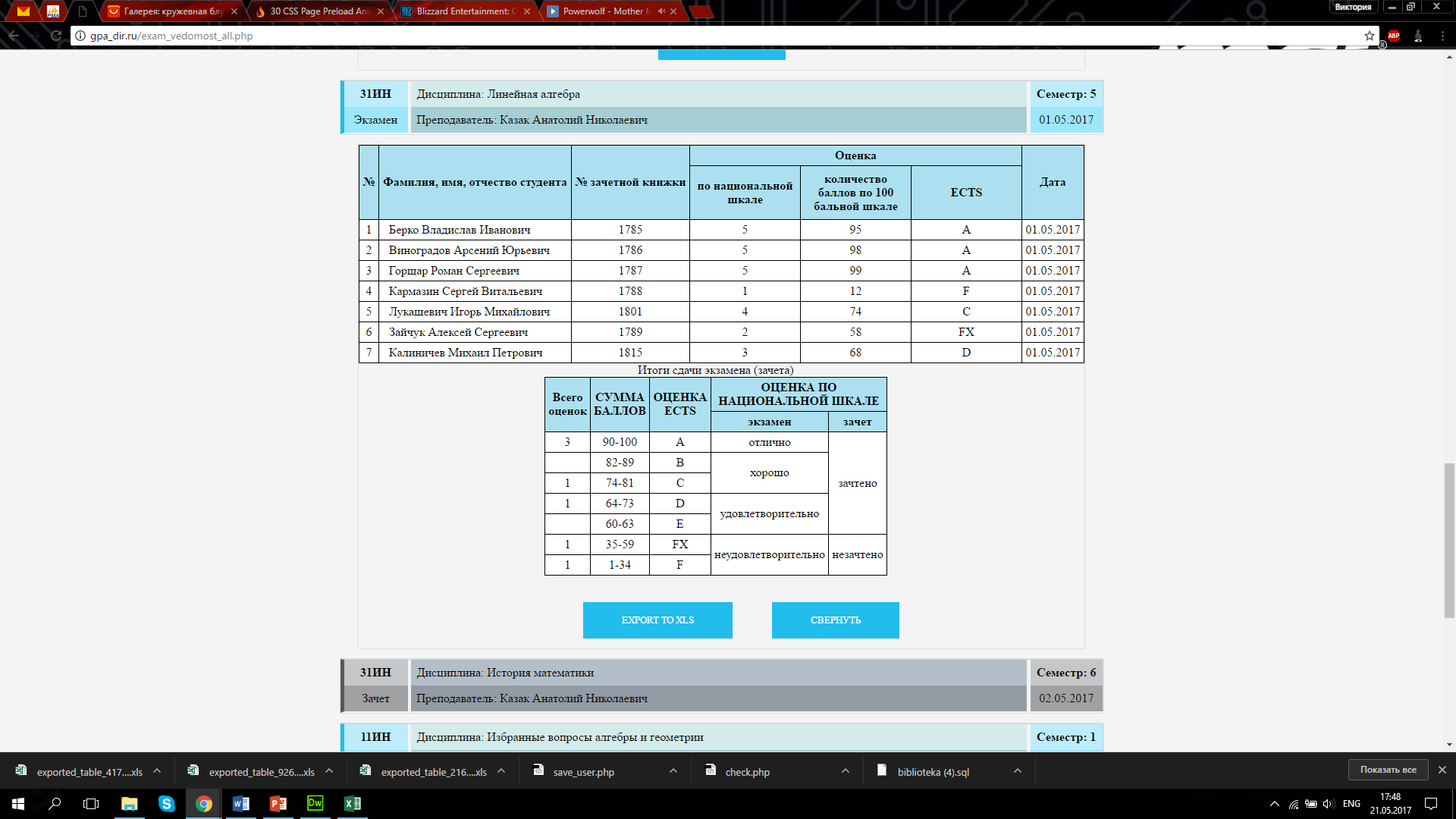


Рис.3.8. Страница просмотра ведомости

В системе реализована возможность экспортировать данные в формат xls. Дополнительные данные в заголовке таблицы выводятся с помощью запросов, чтобы обеспечить требуемый формат ведомости.

Ведение журнала академических групп является также немаловажным для правильной и полной работы системы. Добавлять и редактировать данные может только пользователь с типом «Преподаватель». В личном кабинете выбирается группа. Далее выстраивается список обучающихся данной группы. При выставлении оценки внизу таблицы ведется автоматический подсчет дополнительных сведений: количество человек в группе, средняя оценка по группе, средняя оценка по курсу (по текущей дисциплине). Вид журнала представлен на рис.3.9.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рис.3.9. Журнал текущей успеваемости

Самым важным модулем данной системы является модуль формирования отчетов. В данной системе имеется возможность просмотра следующих отчетов (рис.3.10):

* текущая успеваемость (по курсу, по группе, все группы, по студенту);
* промежуточная успеваемость;
* итоговая аттестация;
* списки обучающихся, отчисленных за указанный период;
* рейтинг обучающихся;
* статистика отчисления обучающихся;
* списки обучающихся, завершивших обучение (по годам выпуска или по группам);
* списки допущенных к сессии;
* списки не сдавших зачеты (экзамены);
* список имеющих академическую задолженность;
* сведения о качестве успеваемости обучающихся;
* сравнение отчетов за прошлые года.

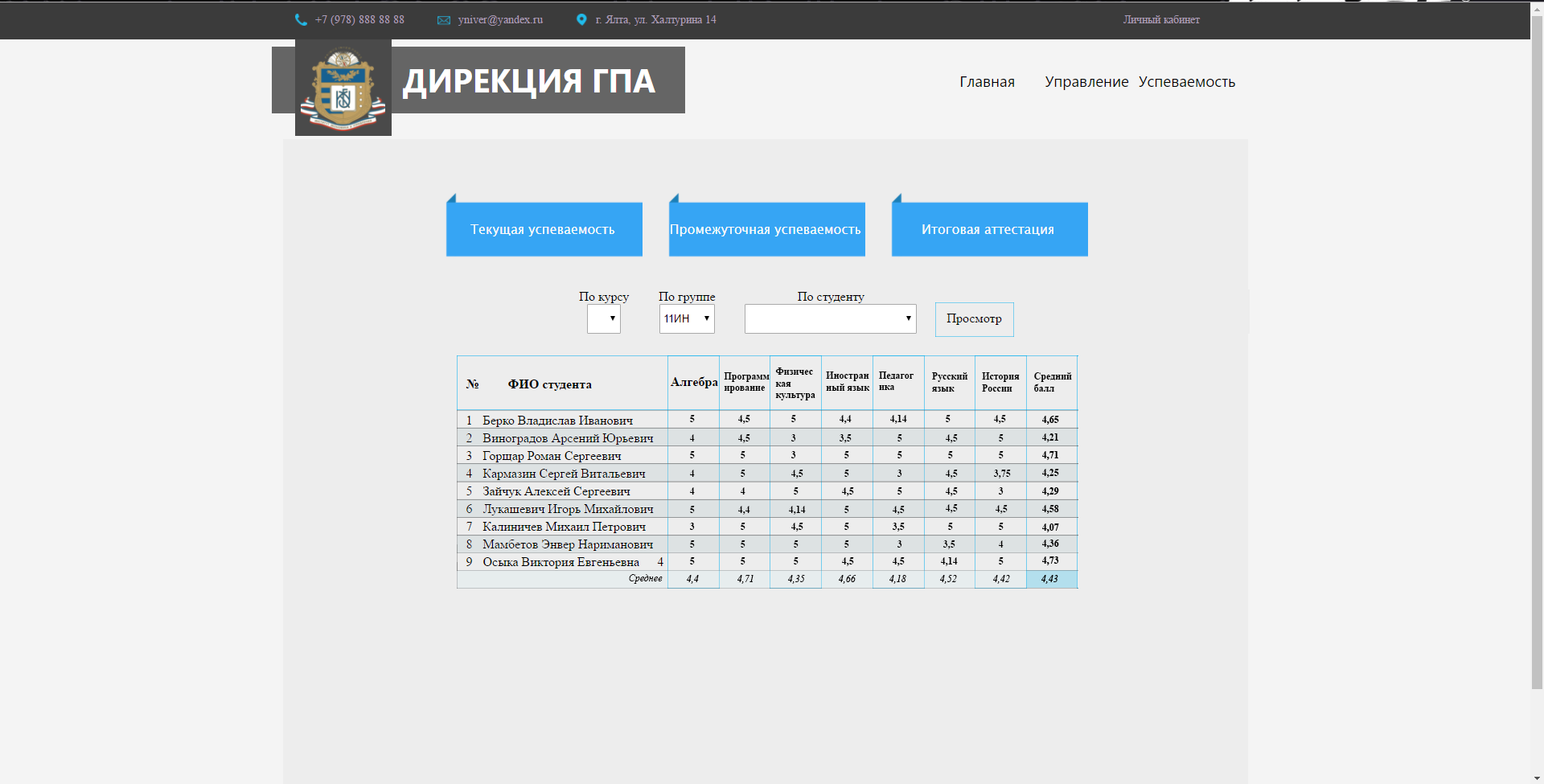


Рис.3.10. Отчет о текущей успеваемости группы

Страницы отчетов имеют одинаковое оформление, но отличаются содержанием. Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплин. Отчеты о текущем контроле включают анализ посещений учебных занятий обучающимися, выполнение форм текущего контроля, и оценку уровня знаний по дисциплине. Результаты запросов можно сохранить в формате xls.

Поля выбора дисциплины и группы взаимозависимы. Выбор в одном поле влияет на данные в другом. Например, выбор студента также зависит от выбранной группы. При нажатии на колонку Всего: пропусков или опозданий, подсчеты будут отображены в процентах от количества занятий.

Для начисления стипендии обучающимся на бюджетной форме необходимо подсчитать средний балл по итогам промежуточной аттестации и исключить тех, кто имеет хотя бы одну удовлетворительную оценку. Для формирования таких сведений предназначена страница отчета На стипендию (рис.3.11).

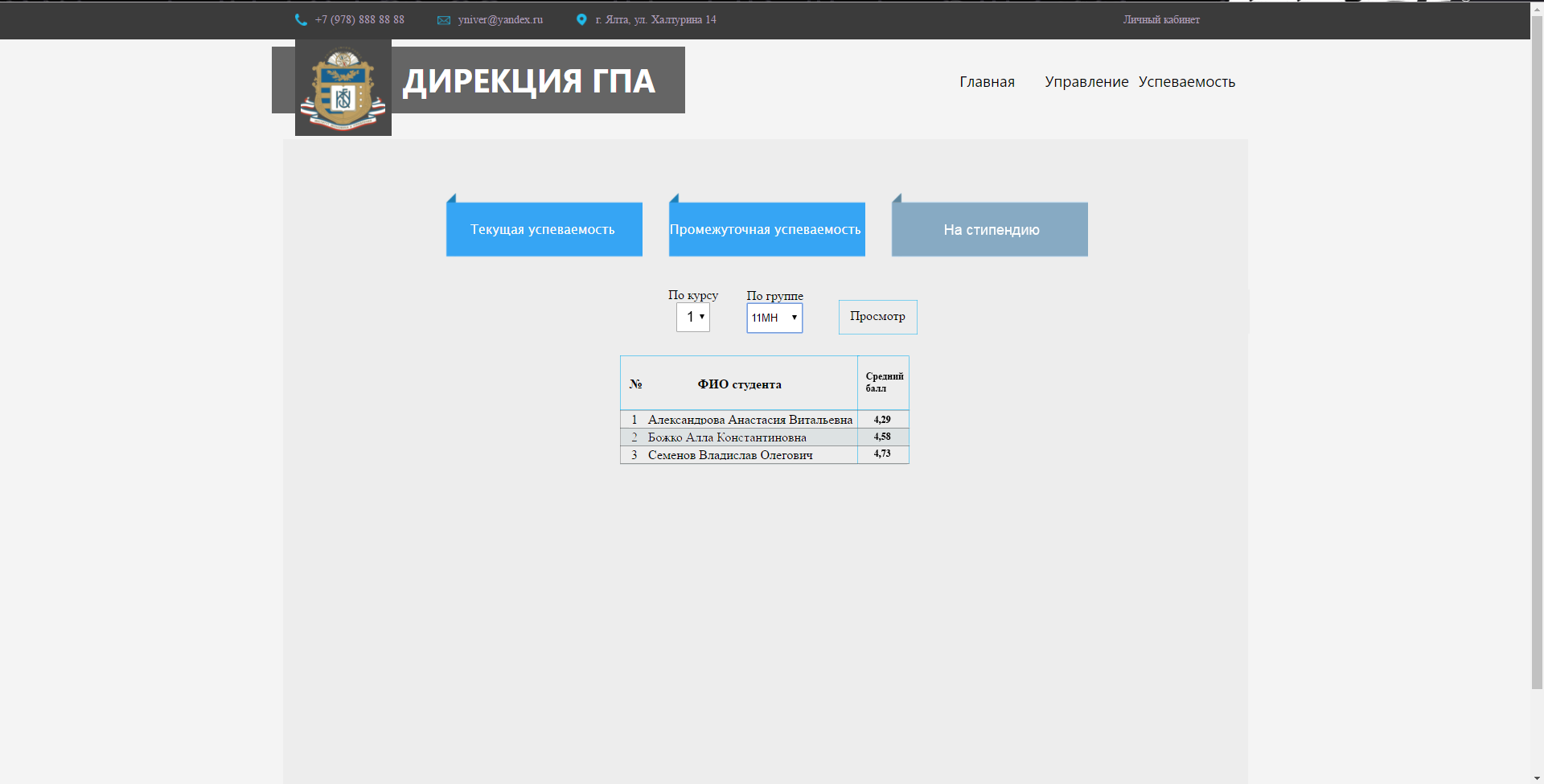


Рис.3.11. Отчет о посещаемости группы по предмету

Данная система имеет многоуровневую архитектуру «клиент-сервер», которая позволяет пользователям обращаться к серверу за необходимой информацией с различных компьютеров. Сервер предоставляет возможность клиенту работать с базой данных. Все принятые запросы от клиента, обрабатываются и направляются в базу. Полученный результат от запроса возвращается клиенту в нужном ему виде.

В процессе реализации информационной системы учета успеваемости обучающихся была выявлена входная информация, необходимая для решения поставленных задач и получения корректных результатов. Перечень и описание входных данных, поступающих в систему представлен в табл.3.1.

Таблица 3.1

Входные данные

|  |  |
| --- | --- |
| Таблица | Входная информация |
| Пользователи | Фамилия, имя, отчество, e-mail, пароль, номер телефона, пол, тип, дата регистрации. |
| Студенты | Номер зачетной книжки, группа, учебный план, статус активности (отчислен или завершил обучение). |
| Преподаватели | Должность, институт, статус активности. |
| Таблица | Входная информация |
| Кафедры | Название, сокращенное название, статус активности. |
| Дисциплины | Название, кафедра, статус активности. |
| Группы | Название, курс, профиль подготовки, статус активности. |
| Направления подготовки | Код, название, кафедра, квалификация, статус активности. |
| Профили подготовки | Код направления подготовки, название, статус активности. |
| Учебные планы | Программа подготовки, профиль подготовки, форма обучения, год начала и конца обучения, Дисциплина, форма контроля, по плану и за семестр: количество зачетных единиц, лекции, лабораторные, практические, потоки, кафедра |
| Экзаменационные ведомости | Группа, семестр, дата сдачи, дисциплина, преподаватель, Студент, оценка по 100 бальной шкале. Статус активности. |
| Журнал академических групп | Группа, дисциплина, преподаватель, дата, тип занятия, тема, студент, оценка/присутствие. |

Выходными данными, получаемыми в результате выполнения различных функций, являются:

1. Отчеты.
2. Информация о кафедрах, дисциплинах и др.
3. Лист журнала учета успеваемости.
4. Листы экзаменационной ведомости.
5. Учебные планы.

# **ЗАКЛЮЧЕНИЕ**

В рамках проекта была разработана информационная система учета промежуточной и итоговой аттестации обучающихся высших учебных заведений Республики Крым. Учет успеваемости обучающихся является сложным и трудоемким процессом, поэтому разработка и внедрение подобной системы, способной автоматизировать централизованное хранение информации и процесс получения статистических данных и отчетов по промежуточной и итоговой аттестации является на сегодняшний день крайне актуальной задачей.

В ходе работы был проведен анализ деятельности дирекции института по учету успеваемости и сформулирована постановка задачи. С применением CASE-средств проведено семантическое моделирование данных. На основе инфологической модели была спроектирована и создана база данных, как основа информационной системы. После сравнительного анализа технологий и инструментальных средств создания информационных систем, существующих на сегодняшний день и используемых разработчиками программного обеспечения, в качестве графического интерфейса был выбран веб-интерфейс. В качестве средств разработки использовались язык разметки гипертекста HTML, формальный язык описания внешнего вида документа CSS, прототипно-ориентированный сценарный язык программирования JavaScript, библиотека JQuery, скриптовый язык общего назначения – PHP, клиент-серверная реляционная система управления базами данных – MySQL.

Разработана многопользовательская система с разграничением прав доступа к информации, а применение архитектуры «клиент-сервер» позволяет разделить логику представления и обработки данных.

В разработанном программном продукте реализованы функции ввода, корректировки, хранения и удаления информации. При работе с системой предусмотрены средства обработки ошибок, а дружественный интерфейс приложения, система подсказок и помощи позволит работать с этой системой даже пользователю, не обладающему высокой квалификацией в области использования информационных систем. Функционал разработанной системы обеспечивает хранение данных о кафедрах ВУЗа, дисциплинах учебного плана, преподавателях, обучающихся, итогах аттестаций, что позволяет получить на основании этой информации электронные варианты журналов учета успеваемости, экзаменационных ведомостей, сформировать необходимые отчеты по различным параметрам.

По материалам проекта были напечатаны статьи совместно с руководителем.

Выходные данные статьи: Филимоненкова Т.Н. Моделирование данных для информационной системы учета успеваемости обучающихся / Т.Н. Филимоненкова, В.Е. Осыка, Р.С. Горщар // Научная дискуссия: вопросы технических наук: сб. ст. по материалам LVI Международной научно-практической конференции «Научная дискуссия: вопросы технических наук». – № 3(43). – М., Изд. «Интернаука», 2017. С. 6-10

(добавь сюда последнюю статью в нашем журнале)

Преимуществами разработанного приложения являются его простота, универсальность, легкая модифицируемость и современные принципы разработки веб-интерфейса, заложенные в его основу.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Белов, В. В. Проектирование информационных систем [Текст] : учебник для студентов, обучающихся по направлению "Прикладная информатика" и другим экономическим специальностям / В. В. Белов, В. И. Чистякова. - 2-е изд., стер. - М. : ИЦ "Академия", 2015. - 352 с. : табл., -схемы, ил. - (Высшее образование. Бакалавриат). - Библиогр.: с. 345-347.
2. Гаврилов, М. В*.* Информатика и информационные технологии: учебник для прикладного бакалавриата / М. В. Гаврилов, В. А. Климов. — 4-е изд., перераб. и доп. — М. : Юрайт, 2017. — 383 с.
3. Заботина, Н.Н. Проектирование информационных систем : учебное пособие / Н.Н. Заботина. - М. : НИЦ ИНФРА-М, 2014. - 331 с. [Электронный ресурс] Режим доступа: http://znanium.com/ catalog.php?bookinfo=454282
4. Информационные системы и технологии управления [Текст] : учебник для студентов вузов, обучающихся по направлениям "Менеджмент" и "Экономика", специальностям "Финансы и кредит", "Бухгалтерский учет, анализ и аудит" / ред. Г. А. Титоренко. - 3-е изд., перераб. и доп. - М. : ЮНИТИ-ДАНА, 2014. - 592 с. : ил. - (Золотой фонд российских учебников). - Библиогр.: с. 572-575. - Краткий слов. терминов и понятий: с. 579-586.
5. Колисниченко, Д. Н. Движок для вашего сайта. CMS Joomla!, Slaed, PHP-Nuke / Д.Н. Колисниченко. — СПб. : ВБХ-Петербург, 2017. — 371 с.
6. Колисниченко, Д. Н. PHP и MySQL. Разработка Web-приложений / Д.И. Колисниченко. — СПб. : ВБХ-Петербург, 2015. — 593 с.
7. Кудрявцев, В. Б*.* Интеллектуальные системы: учебник и практикум для бакалавриата и магистратуры / В. Б. Кудрявцев, Э. Э. Гасанов, А. С. Подколзин. — 2-е изд., испр. и доп. — М. : Юрайт, 2017. — 219 с. — (Бакалавр и магистр. Академический курс).
8. Маклаков, С.В. BPwin и ERwin: CASE – средства разработки информационных систем / С.В. Маклаков [Электронный ресурс] – Режим доступа: https://www.litmir.co/br/?b=429920&p=1#section\_1
9. Маркин, А. В. Программирование на sql в 2 ч. Часть 1: учебник и практикум для бакалавриата и магистратуры / А. В. Маркин. — М. : Юрайт, 2017. — 362 с. — (Бакалавр и магистр. Академический курс).
10. Мартин, Р. Чистый код. Создание, анализ и рефакторинг. Библиотека программиста / Р. Мартин, М. Мартин. — СПб. : Питер, 2016. — 464 с.
11. Мхитарян, В. С. Анализ данных: учебник для академического бакалавриата / В. С. Мхитарян. — М. : Юрайт, 2017. — 490 с. — (Бакалавр. Академический курс).
12. Никсон, Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 / Р. Никсон. — 3-е издание — СПб. : Питер, 2016. — 768 с.
13. Нильсен, Я. Дизайн Web-страниц: анализ удобства и простоты использования 50 узлов [Текст] : пер. с англ. / Я. Нильсен, М. Тахир. - М. : Вильямс, 2002. — 334 c.
14. Новожилов, О. П. Информатика: учебник для прикладного бакалавриата / О. П. Новожилов. — 3-е изд., перераб. и доп. — М. : Юрайт, 2017. — 619 с. — (Бакалавр. Прикладной курс).
15. Подбор цветового решения для веб-сайта / [Электронный ресурс] – Режим доступа: http://www.mistli.ru/podbor-cvetovogo-resheniya-dlya-veb-sayta
16. Пьюривал, С. Основы разработки веб-приложений / С. Пьюривал. - СПб. : Питер, 2015. — 272 с.
17. Филимоненкова, Т.Н. Моделирование данных для информационной системы учета успеваемости обучающихся / Т.Н. Филимоненкова, В.Е. Осыка, Р.С. Горщар // Научная дискуссия: вопросы технических наук: сб. ст. по материалам LVI Международной научно-практической конференции «Научная дискуссия: вопросы технических наук». — № 3(43). — М., Изд. «Интернаука», 2017. С. 6 -10.