

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего
образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет
вычислительной техники

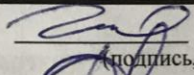
Кафедра
«Вычислительная техника»

Направление подготовки: 09.04.01 «Информатика и вычислительная техника»

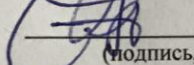
Магистерская программа: «Вычислительные машины, комплексы, системы и сети»

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ
на тему
**СИСТЕМА ХРАНЕНИЯ И ОБРАБОТКИ
ИНФОРМАЦИИ**

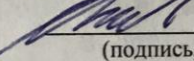
Студент

 Гуляков Сергей Викторович
(подпись, дата) 31.05.2023 (ФИО полностью)

Руководитель

 7.06.23 Р.Н. Федюнин
(подпись, дата) (фамилия, инициалы)

Нормоконтролёр

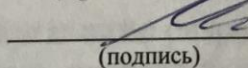
 7.06.23 А.С. Бычков
(подпись, дата) (фамилия, инициалы)

Рецензент

зав. каф. Информатика Пенз. гос. ун-та 08.06.23 Мартышкин А.М.
(должность, место работы) (подпись, дата) (фамилия, инициалы)

Работа допущена к защите (протокол зае. кафедры от 02.06.2023г. № 14)

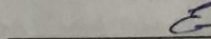
Заведующий кафедрой

 М.А. Митрохин
(подпись) (фамилия, инициалы)

Работа защищена с отметкой отлично (протокол заседания ГЭК от

9.06.2023г. № 8)

Секретарь ГЭК

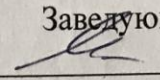
 Е.И. Гурин

Пенза 2023

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет
вычислительной техники

Кафедра
«Вычислительная техника»

«Утверждаю»
Заведующий кафедрой
 М.А. Митрохин
« 29 » 11 2021г.

**ЗАДАНИЕ
НА МАГИСТЕРСКУЮ ДИССЕРТАЦИЮ**

1. Студент Гуляков Сергей Викторович гр. 21BВВм1 факультета вычислительной техники.

2. Тема работы «Система хранения и обработки информации»

Тема утверждена приказом ПГУ №924/о от "29" ноября 2021 г.

3. Руководитель работы к.т.н., доцент кафедры ВТ ПГУ Федюнин Р.Н.

4. Задание на работу (назначение разработки, исходные данные и т.п.)

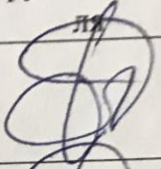
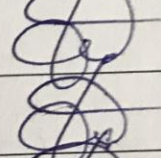
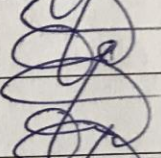
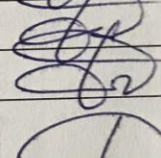
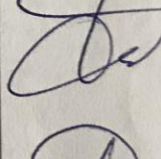
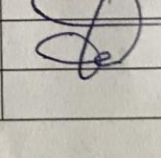
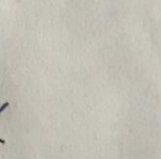
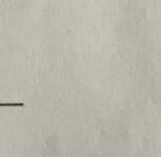
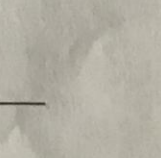
Программное приложение должно быть реализовано в формате клиент-сервер на базе паттерна "модель-вид-контроллер". Основными функциями программного обеспечения должны быть следующие возможности: создание, обновление, удаление, чтение и запись данных по маршруту клиент-серверная часть приложения база данных.

Выше описанная модель должна быть опробованна на примере автоматизированной системы контроля успеваемости учащихся.

5. Перечень подлежащих разработке вопросов

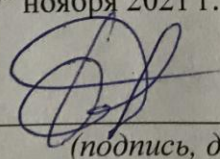
Анализ стека технологий для реализации клиент-серверных приложений. Проектирование и реализация базы данных, проектирование и реализация интерфейса, реализация бизнес логики. Тестирование и получение результатов на базе виртуального сервера.

6. Календарный график выполнения работы

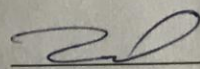
№ п/п	Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя
1.	Анализ существующих систем хранения и обработки информации	5%	20.12.2021	
2.	Анализ средств разработки web-приложений	5%	20.12.2021	
3.	Анализ существующих СУБД и средств разработки БД	5%	20.06.2022	
4.	Анализ способов тестирования клиент-серверных приложений	5%	20.06.2022	
5.	Разработка серверной части системы	10%	20.12.2022	
6.	Разработка БД	20%	20.12.2022	
7.	Разработка web-приложения	30%	20.02.2023	
8.	Комплексная отладка системы хранения и обработки информации на примере реализованной системы контроля успеваемости учащихся посредством тонкого клиента	10%	20.05.2023	
9.	Оформление документации	10%	31.05.2023	

Дата выдачи задания "29" ноября 2021 г.

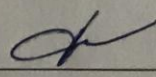
Руководитель работы

 Ферошкин Р.В.
(подпись, дата)

Задание к исполнению принял студент


(подпись, дата)

Работу к защите допустить
Декан факультета

 А.Р. Гусев
(подпись, дата)

РЕФЕРАТ

Пояснительная записка содержит 127 листов, 34 рисунка, 34 таблицы, 6 использованных источников, 2 приложения.

Объектом исследования является система хранения и обработки информации, на примере автоматизированной системы контроля успеваемости учащихся.

Цель работы - разработка программного обеспечения системы контроля успеваемости учащихся посредством тонкого клиента.

Разработка велась на языке PHP.

В процессе разработки был проведен анализ требований к разрабатываемому продукту, анализ существующих современных средств проектирования программных продуктов, анализ аналогичных систем, изучены способы сетевого взаимодействия.

В результате выполнения магистерской работы было разработано программное обеспечение, осуществляющее выставление оценок в электронный журнал учащихся, посредством авторизации каждого преподавателя для дальнейшего просмотра, анализа и контроля родителей с помощью Интернет - приложения. Осуществлено функциональное тестирование разработанного приложения, которое показало корректность его работы.

					ПГУ1.09.04.01.03.001 ПЗ							
Изм.	Лист	№ докум.	Подпись	Дата	Система хранения и обработки информации Пояснительная записка			Лит.	Лист	Листов		
Разраб.		Гуляков С.В.								2	127	
Провер.		Федюнин Р.Н.						ФВТ, каф. ВТ гр. 21ВВВм1				
Н. Контр.		Бычков А.С.										
												
Утверд.		Митрохин М.А.										

СОДЕРЖАНИЕ

РЕФЕРАТ.....	5
ПЕРЕЧЕНЬ ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ.....	5
ВВЕДЕНИЕ.....	6
1 ПРОГРАММНЫЕ СРЕДСТВА АДМИНИСТРИРОВАНИЯ СИСТЕМЫ КОНТРОЛЯ УСПЕВАЕМОСТИ УЧАЩИХСЯ.....	8
1.1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ.....	8
1.1.1 Цели и задачи магистерской диссертации.....	10
1.1.2 МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ПРОЕКТА.....	13
1.2 АНАЛИЗ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ АНАЛОГИЧНОГО НАЗНАЧЕНИЯ.....	15
1.3 ОБЗОР ТЕХНОЛОГИЙ ВЕБ-ПРИЛОЖЕНИЙ	18
1.3.1 ОБЗОР ВЕБ-СЕРВЕРА APACHE.....	18
1.3.2 ОБЗОР ЯЗЫКА ПРОГРАММИРОВАНИЯ PHP.....	21
2 МЕТОДЫ И СРЕДСТВА РЕШЕНИЯ ЗАДАЧИ.....	23
2.1 ТРЕБОВАНИЯ К ФУНКЦИОНАЛЬНОМУ НАЗНАЧЕНИЮ ПРИЛОЖЕНИЯ	23
2.2 АНАЛИЗ ТРЕБОВАНИЙ К ИНТЕРФЕЙСУ ПРИЛОЖЕНИЯ	24
2.2.1 ТРЕБОВАНИЯ К СРЕДСТВАМ РАЗРАБОТКИ.....	24
2.3 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	26
2.3.1 ПРОЕКТИРОВАНИЕ СТРУКТУР ДАННЫХ	26
2.3.1.1 КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ.....	26
2.3.2 ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ	27
2.3.3 ДАТАЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ.....	32
2.3.4 ИНФОЛОГИЧЕСКАЯ МОДЕЛЬ	43
2.4 ПРОЕКТИРОВАНИЕ СТРУКТУРЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.....	45
2.4.1 ПРОЕКТИРОВАНИЕ АЛГОРИТМОВ	47
2.5 ПРОЕКТИРОВАНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА	48
2.5.1 РЕАЛИЗАЦИЯ.....	69

3	ТЕХНОЛОГИЯ РАЗРАБОТКИ И ОТЛАДКИ WEB-ПРИЛОЖЕНИЙ В СРЕДЕ NetBeans .	
		71
3.1	Создание проекта в NetBeans	71
3.2	Отладка кода с помощью XDebug	75
	ЗАКЛЮЧЕНИЕ	79
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	80
	ПРИЛОЖЕНИЕ А	81
	ПРИЛОЖЕНИЕ Б.....	121

ПЕРЕЧЕНЬ ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ

В настоящей работе применяются следующие термины с соответствующими определениями

БД - база данных

НФ - нормальная форма

ПЗ - пояснительная записка

ПИ - пользовательский интерфейс

ПК – персональный компьютер

ПО – программное обеспечение

ТМО –теория массового обслуживания

ПЭВМ - персональная электронно-вычислительная машина

СУБД - системы управления данными

ЭВМ - электронно-вычислительная машина

РС –personalcomputer

ВВЕДЕНИЕ

В последнее время в сфере образования наметилась тенденция на изменение системы ведения журналов успеваемости.

Внедрение системы контроля успеваемости – Электронного дневника позволяет образовательному учреждению автоматизировать составление школьной отчетности, оперативно получать различные статистические срезы с целью повышения качества управленческой деятельности. Школьникам стать более ответственными и открытыми перед родителями, а родители получают возможность принять непосредственное участие в образовательном процессе ребенка. В конечном итоге благодаря электронному дневнику повышается успеваемость и посещаемость школьников, тем самым, повышается престиж школы.

Основными задачами магистерской диссертации являются: вовлечение образовательных учреждений в процесс электронизации хранения и обработки данных; постепенный переход от стандартных бумажных форм обеспечения учебного процесса на электронные: организация ведения электронного дневника учащегося, электронного журнала успеваемости; максимальное высвобождение преподавательского состава от ручного механического труда.

Целью проектирования является реализация программного продукта, который смог бы обеспечить выполнение функций контроля успеваемости и даже расширить их. Изучив системы аналогичного назначения, были выделены основные направления развития данного программного обеспечения. Основными функциями системы являются функции автоматизации заполнения школьных журналов и тем занятий, выставления оценок учащимся, информирование учащихся посредством объявлений, выдача домашнего задания, анализ успеваемости, представление статистической информации. Компоненты статистики позволяют определять лучших учащихся, выявлять

успеваемость в классах. Они призваны помогать как преподавательскому составу, так и учащимся и административной части учебного заведения. Статистика представлена в удобной для пользователя форме: в виде таблиц. Многие разработчики аналогичных систем в основном пренебрегают построением подсистемы обработки статистической информации. При создании и реализации магистерской диссертации были учтены все функциональные требования потенциальных пользователей системы.

1 Программные средства администрирования системы контроля успеваемости учащихся

1.1 Анализ предметной области

В отличие от высших учебных заведений школа имеет свою специфику, которая позволяет раскрыть важные идеи электронных журналов и переосмыслить их роль в образовательном процессе.

Основной аргумент, который приводят в школах противники подобных нововведений – дополнительная неоплачиваемая нагрузка для учителей. В первую очередь это связано с тем, что в школах выставляется на порядок больше текущих оценок, а перепоручить ученикам работу по внесению информации в электронный журнал не всегда возможно. В совокупности с хронической перегруженностью учителей в большинстве школ и низким уровнем зарплат это действительно создает в педколлективе атмосферу неприятия новой идеи. Кроме этого могут накладываться проблемы технического оснащения школы (недостаточное количество ПК, нестабильный доступ в Интернет и прочее), низкий уровень владения ПК у педагогов, сложность установки и настройки программного обеспечения, психологические проблемы (беспокойство по поводу взлома системы, несанкционированного доступа со стороны учеников, потери информации вследствие технических сбоев и прочее).

Чтобы найти приемлемое решение описанных проблем, необходимо переосмыслить цели внедрения электронных журналов. Основными мотивами для администрации школы чаще всего являются оперативное информирование родителей об успеваемости их детей, автоматизированный расчет показателей учебного процесса и создание определенного имиджа учебного заведения. Можно упомянуть и мнение председателя партии “Единая Россия” Дмитрия Медведева по этому поводу: “Во всяком случае, электронный журнал точно можно ввести. Это будет способствовать обучению компьютерной

грамотности тех педагогов, которые боятся компьютеров” (по материалам РИА Новости).

Но, как справедливо отмечают консервативно настроенные педагоги, эти факторы не играют решающей роли в школе. Во-первых, есть дневники, которые собственно и предназначены для передачи информации родителям, во-вторых, значительная часть родителей, к сожалению, не проявляет интереса к оценкам своих детей. Но, самое главное, оценка практически полностью утратила свою важнейшую роль – быть стимулом в учебе. Дети и их родители становятся равнодушными к тем оценкам, которые они получают. И это, к слову, корень многих проблем в современной школе. Вследствие сказанного, основной целью внедрения электронных журналов в школе должно стать поднятие «имиджа» оценки и укрепления ее роли, как мощного стимула в учебном процессе. Такой эффект может быть достигнут в случае создания долговременной открытой базы оценок и введение на ее основе некоего рейтинга учащихся, хотя бы на уровне отдельных школ. Осознание детьми и родителями того факта, что их результаты публикуются в сети Интернет в открытом доступе, заставит относиться к оценкам более ответственно. Природу такого стимула легче понять, если провести аналогию, например, со спортом, где все результаты и рейтинги спортсменов открыты и служат зачастую смыслом спортивного роста. В полной мере ощутить такой результат можно будет со временем, когда база оценок наберет свою «критическую массу».

Конечно, чтобы такой принцип не противоречил законам о защите персональных данных (например, на территории РФ действует закон N 152-ФЗ) школе необходимо взять добровольное письменное согласие родителей на публикацию оценок их детей. Как показывает практика, большинство понимают такую роль электронных журналов и принимают положительное решение. Для остальных детей нужно обеспечить ограниченный доступ к информации или вообще не вносить их оценки в электронную базу.

С учетом сказанного, можно предложить простое решение вопроса о дополнительной нагрузке для учителей. В электронные журналы не обязательно вносить все текущие оценки, достаточно только наиболее важные, например, четвертные и годовые. Это снимет большинство проблем, но позволит достичь основных целей. В заключение можно отметить, что электронные журналы необходимо рассматривать не как новомодный способ представления информации об успеваемости, а как мощное средство повышения успеваемости и уровня образования в целом на уровне школы.

1.1.1 Цели и задачи магистерской диссертации

Целью диссертации является разработка программного обеспечения системы контроля успеваемости учащихся.

Разрабатываемая система рассчитана, на преподавателей оценивающих учащихся по бальной системе. Им важно получать статистику успеваемости, наглядно видеть промежуточные и итоговые результаты, как всего класса, так и конкретно одного ученика.

Также разрабатываемую систему могут применять учащиеся и их родители, чтобы посмотреть рейтинг успеваемости и оценить свой уровень знаний.

К основным функциям программ, реализующих работу подсистемы, администраторской части программного обеспечения администрации учебного заведения относятся:

- регистрация школы в системе и получение личного кода доступа к функциям администрирования школой,
- авторизация,
- просмотр статистических данных школы,
- формирование списков преподавателей и привязка их к школьным предметам,
- формирование учебных классов,
- добавление учащихся,

- формирование учебных дисциплин,
- формирование средней оценки класса,
- построение графика успеваемости.

К основным функциям программ, реализующих работу подсистемы, администраторской части программного обеспечения преподавателями учебного заведения относятся:

- авторизация,
- выбор только тех классов, в которых преподаватель ведет дисциплины,
- добавление объявлений в классы,
- выставление оценок по своим предметам,
- выдача домашнего задания учащимся.

К основным функциям программ, реализующих просмотр успеваемости учениками, относятся:

- авторизация,
- просмотр всех учеников класса,
- обмен сообщениями между одноклассниками,
- просмотр объявлений от преподавателей,
- просмотр диаграммы успеваемости и количество полученных оценок,
- получение данных о преподавателях, которые ведут занятия в классе где учится ученик.

Для эксплуатации, приложение может быть установлено как на отдельном компьютере, так и в локальной компьютерной сети.

Для развертывания веб-сервера Apache необходим PC совместимый компьютер со следующими техническими показателями:

- процессор с тактовой частотой не ниже 2000 МГц,
- оперативная память не менее 512 Мб (рекомендуется 1 Гб и более),
- жесткий диск с объемом свободного места не менее 5 Гб,

- монитор SVGA, поддерживающий разрешение не менее 800x600.

Веб-сервер Apache, популярный благодаря своей прозрачности для программистов и администраторов и не в последнюю очередь благодаря бесплатности, реализован под все более-менее популярные операционные системы. Его настройки для всех ОС одинаковы, различаются только пути, где хранятся конфигурационные файлы. На одном сервере может быть запущено несколько экземпляров Apache, прослушивающих разные адреса и/или порты. Один экземпляр может обслуживать несколько сайтов. Веб-сервер Apache настолько популярен, что постоянно обрывает новую функциональность. О степени популярности можно судить либо по статистике, либо по конкретным фактам. Фактов популярности Apache можно привести два. Во-первых, его наличие в качестве веб-сервера по умолчанию в большинстве UNIX-подобных операционных систем. Во-вторых, такой гигант в области ИТ как компания Oracle приняла его в состав своего продукта Oracle Application Server 10g, где он играет одну из ключевых ролей.

Для развертывания СУБДMySQL необходим IBM совместимый компьютер со следующими техническими показателями:

- процессор с тактовой частотой не ниже 2500 МГц,
- оперативная память не менее 512Мб (рекомендуется 1Гб и более),
- жесткий диск с объемом свободного места не менее 200 Мб,
- монитор SVGA, поддерживающий разрешение не менее 800x600.

MySQL является решением для малых и средних приложений. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Сервер СУБД MySQL должен функционировать под управлением операционной системы Microsoft Windows XP, Microsoft Windows Server 2003, Microsoft Windows Server 2008, Microsoft Windows Server 2008 R2, Microsoft Windows Server 2012, Microsoft Windows Server 2012 R2, Microsoft Windows

Server 2016, Microsoft Windows Server 2019, Microsoft Windows Vista, Microsoft Windows 7, Microsoft Windows 8/8.1, Microsoft Windows 10, Microsoft Windows 11.

Для работы клиентского приложения, необходим IBM совместимый компьютер со следующими техническими показателями:

- процессор с тактовой частотой не ниже 500 МГц,
- оперативная память не менее 128 Мб,
- монитор SVGA, поддерживающий разрешение не менее 800x600,
- принтер.

Клиентские приложения должны функционировать под управлением операционной системы WindowsNT/2000/XP/7/8/8.1/10.

Для использования системы на серверной и клиентской стороне необходимо наличие браузера(желательно FireFox версии не ниже 3.6).

1.1.2 Математическая модель проекта

Теория массового обслуживания (ТМО)- это раздел теории вероятностей, посвященный изучению систем обработки заявок, имеющих случайную природу. В ТМО под заявкой понимают объект, который поступает в систему и требует времени на его обработку. Под сервером обычно понимается оператор, канал связи или другие ресурсы, занимаемые заявками и блокируемые на время обработки.

Аппарат теории массового обслуживания разрабатывался со времени появления телекоммуникаций как отрасли. Научные работы, положившие основание ТМО, принадлежат Агнеру Краупу Эрлангу — сотруднику Копенгагенской телефонной компании. Формулы Эрланга используются повсеместно при расчетах достаточной пропускной способности сети и максимальной нагрузки на сервер.

Система массового обслуживания характеризуется тремя основными параметрами: интенсивностью входящего потока, производительностью сервера и числом серверов в системе. Если бы заявки поступали в систему и

покидали ее в строго определенные моменты, то система работала бы без очередей. В реальности возможно одновременное появление многих заявок в системе. В этом случае они становятся в очередь, откуда другое название ТМО: теория очередей.

Основная задача ТМО - определить вероятность того, что заявка будет обслужена вовремя. Обычно в ТМО рассматривается максимальное время ожидания заявки в очереди, но можно рассчитать и полное время на обработку заявки. На практике интерес представляет обратная задача: рассчитать число серверов, необходимых для поддержания требуемого уровня обслуживания.

В нашем случае определим максимальное число каналов, способных перенести пиковую нагрузку (например, в период сессии) по формуле Эрланга:

$$p^k = \frac{a^k / k!}{\sum_{k=0}^n \frac{a^k}{k!}} \quad 0 \leq k \leq n$$

При частоте запросов на сервер, равным 1000 в секунду и среднем времени обработки запроса равным 1 секунде таблица распределения вероятности отказа будет иметь вид, таблица 1.1.

Таблица 1.1 – Вероятность отказа

Вероятность отказа	Количество каналов
90 %	101
80 %	201
70 %	301
60 %	401
50 %	501
40 %	602
30 %	703
20 %	804
10 %	909
5 %	966
4 %	979
3 %	993
2 %	1009
1 %	1029

Отсюда следует, что для нормальной работы в период максимальных нагрузок системе необходимо около 1029 каналов обработки данных. Система располагает такими ресурсами, так как будет размещена на веб-сервере с мощным серверным оборудованием.

1.2 Анализ программного обеспечения аналогичного назначения

К аналогам программного продукта, реализуемого в магистерской диссертации, можно отнести программы автоматического контроля успеваемости учащихся. Рассмотрим примеры программ данных тематик.

Интерактивная система "Электронный школьный журнал и дневник школьника на основе web-технологий" компании SoftLink.

Продукт создавался разработчиками компании SoftLink. Стоит отметить, что данная система предполагает использование только в одной школе, используя свой или сторонний сервер для реализации серверного программного обеспечения. Отсюда следует, что только одна школа может использовать данную систему. Программа "Электронный школьный журнал и дневник школьника на основе web-технологий" является Интернет-приложением и может быть доступна по адресу <http://penza.e-zurnal.ru>. (рисунок 1.1).

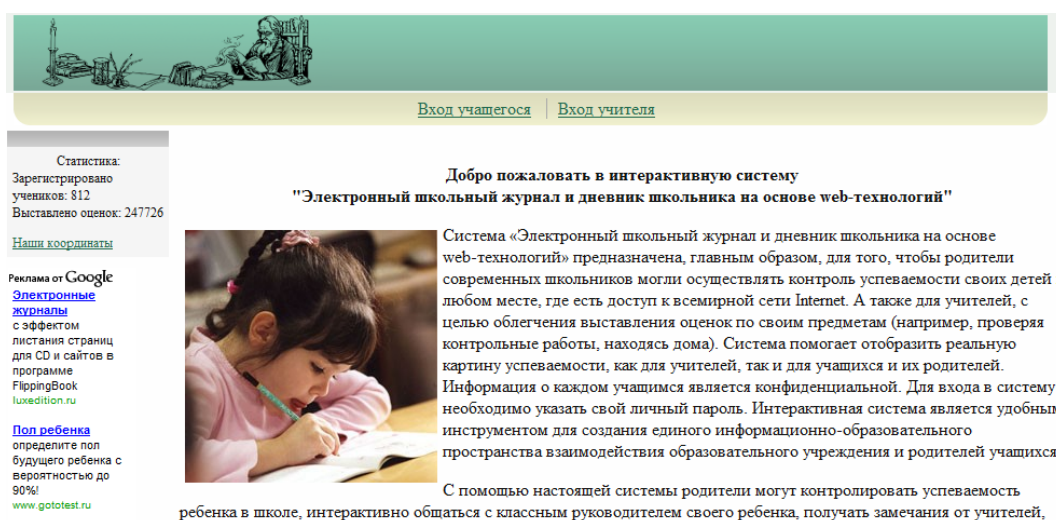


Рисунок 1.1 –Интерактивная система "Электронный школьный журнал и дневник школьника на основе web-технологий".

Для авторизации пользователя используется название класса и пароль (для студентов), логин и пароль (для преподавателей).

В системе реализуются следующие функции:

- авторизация студентов и преподавателей,
- выставление баллов по дисциплинам,
- просмотр успеваемости учащимися и родителями учащихся.

В качестве недостатка следует отметить отсутствие статистической информации и её графического представления. Примитивный пользовательский интерфейс и интерфейс администрирования системы.

Серверную часть системы, разработанную в данном проекте, предполагается разместить на одном сервере, с возможностью использования ее множеством школ.

Система электронных журналов успеваемости Education Management System.

Данная система представляет собой сервис электронных журналов с созданием и ведением электронных журналов успеваемости, sms и mail-рассылкой. Для работы системы предполагается выделение отдельного домена для каждой школы.

Любое учебное заведение или отдельный преподаватель могут создать и вести в системе EMSY любое количество электронных журналов для различных учебных групп (классов), использовать различные системы оценивания (5-ти, 12-ти или 100-бальную), самостоятельно определять виды оценок (текущие, аттестационные, итоговые, экзаменационные и т.д.), учитывать пропуски занятий учащимися по уважительной и без уважительной причинам, использовать сводные ведомости, группировку оценок по предметам или учащимся, фильтры оценок и многое другое.

Учебное заведение может использовать политику открытого или закрытого доступа к электронному журналу. Открытый доступ позволяет любому посетителю сайта просмотреть любые оценки. При включенной

политике закрытого доступа для просмотра оценок необходимо знать коды доступа, которые индивидуальны для каждого студента (учащегося). Возможен комбинированный вариант, при котором сами студенты (учащиеся) или администрация в индивидуальном порядке определяют политику доступа.

Продукт разрабатывался командой разработчиков EMSY и доступен по адресу <http://www.emsy.org> (рисунок 1.2).

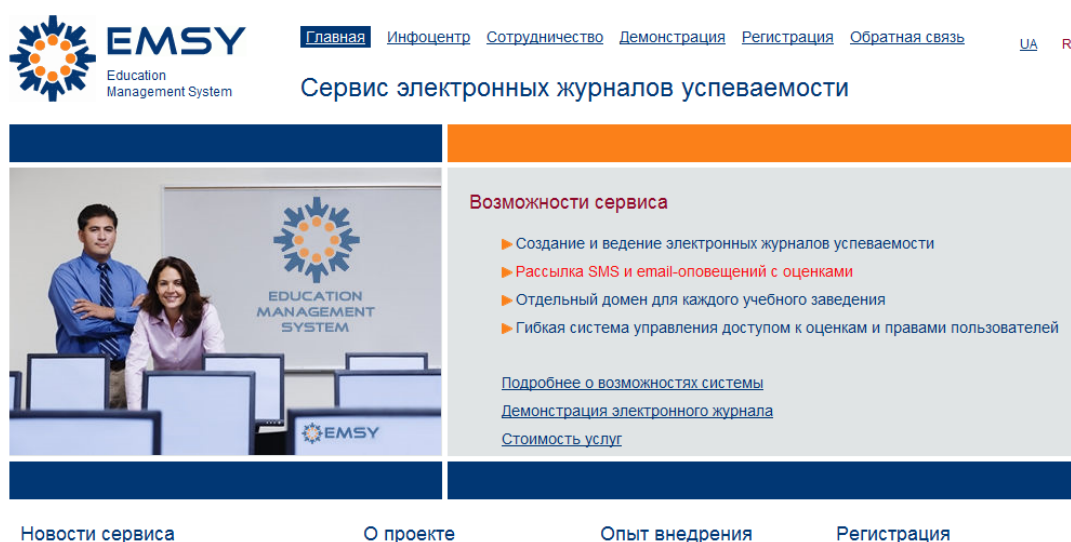


Рисунок 1.2 –Система электронных журналов успеваемости Education Management System.

Данная система регламентирует следующие возможности программы:

- в журналы заносятся аттестационные, итоговые(семестровые) и экзаменационные оценки всех студентов техникума дневной формы обучения,
- оценки в журналы вносят старосты групп на основе оценок обычных журналов, после чего проверяются и заверяются(подписываются) зав.отделением,
- до 10 числа каждого месяца в учебный период, родителям студентов рассылаются email и SMS-оповещения с информацией об успеваемости их студентов за прошедший месяц,
- все оценки, которые вносятся в электронные журналы, доступны для просмотра любому посетителю сайта.

Архитектура системы позволяет выделить для учебного заведения отдельный высокопроизводительный сервер, разместить его на любой технической площадке, использовать отдельную базу данных и независимую версию программного кода (последнее позволяет выполнять индивидуальные доработки системы). Для предотвращения потери данных из-за сбоев в работе системы каждый день автоматически создается резервная копия базы данных.

Недостатками данной системы является абонентская плата, коротая взимается с родителей учащихся за предоставление данного сервиса. Большая нагрузка на сервер в связи с дополнительными ненужными сервисами, при большой посещаемости это может отразиться на производительность работы системы.

1.3 Обзор технологий веб-приложений

1.3.1 Обзор веб-сервера Apache

За последние несколько лет Интернет стремительно проник почти во все сферы человеческой жизни и деятельности. В сети, которая изначально задумывалась для нужд Министерства обороны США, нашлось место всем: и самому этому министерству, и государственным и коммерческим организациям, и частным лицам, и вообще всему, что только можно вообразить. В целом - это огромная, сложная, но вместе с тем довольно стабильно работающая система со множеством сервисов, программ, разработок и т.п. Но многие сегодня думают, что Интернет – это WWW и Web-браузеры. И это не случайно. WWW, в отличие от e-mail или FTP, позволяет не просто передавать файлы или текстовые письма, а, скажем, показывать картинки, передавать в реальном времени звук и даже видео. При всем этом вам ненужно ничего скачивать к себе на компьютер: вы просто используете весь мир как одну большую сеть. Необходимостью является, пожалуй, лишь хорошая скорость передачи данных.

Web-сервер - это программа, обрабатывающая сообщения, которые приходят на 80-й порт (стандартная настройка; можно конечно, настроиться и

на любой другой порт), и работающая с протоколом HTTP (Hypertext Transfer Protocol). Именно этот протокол является основным для WWW. Он представляет собой набор правил для обмена данными и основан на принципе «запрос-ответ». Запрос идет от клиента к серверу и содержит служебную информацию о типе запроса (данные, заголовок, форма), заголовок запроса (допустимые типы файлов, авторизация, версия клиента, адрес, где была активирована ссылка на данный ресурс, и сам адрес) и запрос данных. Ответ идет от сервера к клиенту. В нем находится служебный код (число), показывающий состояние обработки запроса, ответный заголовок (версию сервера, дату, длину и тип данных) и сами данные.

Сейчас в мире существуют довольно много Web-серверов (несколько десятков), но популярностью пользуются лишь несколько. Одним из самых популярных веб-серверов является Apache от ASF (Apache Software Foundation).

Без преувеличения можно сказать, что это самый распространенный Web-сервер в мире. Сейчас имеются версии фактически для всех известных платформ и операционных систем. Его несомненными достоинствами являются надежность, исключительная производительность и огромный набор функций и дополнительных модулей. Но все же «изюминкой» этого сервера является свободное бесплатное распространение. Это дает возможность, помимо экономии денег, быстро исправлять ошибки и вносить в код программы необходимые дополнения. Надо отдать должное тем, кто занимается созданием Web-сервера Apache, эти люди безвозмездно и, главное, очень быстро справляются с замеченными ошибками. Кстати, отмечу, что таким человеком сможет стать любой, желающий внести свой вклад в развитие этого сервера. Версия 1.3.x считается одной из самых стабильных и быстрых среди всего семейства Apache. Основными свойствами Apache являются поддержка кросс-платформ, протокол HTTP/1.1, модульная структура, защита, запись log-файлов. Кроме стандартной поставки имеется большое количество модулей,

которые расширяют возможности Apache. Свободное распространение и открытый код позволяют создавать наиболее защищенные модули. В этом с Apache трудно соревноваться любому коммерческому серверу. Но, несмотря на все эти достоинства, есть и серьезный пробел: у Apache нет красивой программы установки и управления. Во время установки приходится иметь дело с командной строкой. Более того, при установке под UNIX иногда приходится даже вносить коррективы в include-файлы. Конечно, есть подробное описание того, что и как необходимо изменить, однако, на первый взгляд это кажется не очень удобным. При работе приходится иметь дело не с оконной программой, а с текстовыми конфигурационными файлами.

Но такой нетипичный для серьезного продукта стиль не делает Apache менее популярным, и на рынке он является самым серьезным конкурентом для всех WWW-серверов.

Краткие характеристики Web сервера Apache:

- последняя версия: 2.4.57,
- последняя официальная версия: 2.4.X,
- бесплатный, открытый код,
- операционные системы: NetBSD, Digital UNIX, BSDI, AIX, OS/2, SCO, HP-UX, Novell NetWare, Macintosh, Be OS, Windows NT, Linux, Windows 95, FreeBSD, Windows 98, IRIX, Solaris, Microsoft Windows XP, Microsoft Windows Server 2003, Microsoft Windows Server 2008, Microsoft Windows Server 2008 R2, Microsoft Windows Server 2012, Microsoft Windows Server 2012 R2, Microsoft Windows Server 2016, Microsoft Windows Server 2019, Microsoft Windows Vista, Microsoft Windows 7, Microsoft Windows 8/8.1, Microsoft Windows 10, Microsoft Windows 11
- может создавать несколько log-файлов, под Windows запускается как сервис и/или программа, под UNIX может запускаться из inetd, допускает настройку на несколько портов,

- поддерживает Windows CGI, HTTP/1.1, в том числе и HTTP/1.1 PUT; есть функция автоматического ответа при изменении документа; поддерживает Microsoft ISAPI;
- возможен запрет доступа с конкретных адресов, к конкретным документам, запрет запуска CGI скриптов, доступ конкретных пользователей; допускаются изменения без перезапуска сервера;
- поставляется вместе с полным исходным кодом; поддерживает другие протоколы (ftp, telnet); имеет настройку на пользовательские директории; содержит модуль проху.

Данные Wikipedia (https://ru.wikipedia.org/wiki/Apache_HTTP_Server) для сервера Apache:

- процент на рынке Web-серверов — 33,56%,
- количество — более 304 млн сайтов.

Рейтинг (по пятибалльной системе):

- надежность – 5,
- производительность – 5,
- простота использования – 3,
- техническая поддержка - 4,5.

1.3.2 Обзор языка программирования PHP

PHP - интерпретируемый скриптовый язык программирования, созданный для генерации HTML-страниц на веб-сервере и работы с базами данных. В области веб-программирования PHP является на сегодняшний день одним из самых распространённых технологий (наряду с Perl, ASP/.NET и Python) благодаря простоте, скорости выполнения и богатой функциональности. PHP распространяется свободно. Синтаксис языка похож на синтаксис C++. PHP поддерживается подавляющим большинством поставителей сетевого хостинга.

Название "PHP" представляет собой самоповторяющуюся (рекурсивную) аббревиатуру и расшифровывается как "PHP: Hypertext Preprocessor", или

"Personal Home Page". PHP был создан в качестве надстройки над Perl для облегчения разработки веб-страниц. За свою жизнь PHP значительно изменялся. Одной из сильнейших сторон PHP является возможность расширения ядра. Интерфейс написания расширений привлек к PHP множество сторонних разработчиков, работающих над своими модулями, что дало PHP возможность работать с огромным количеством баз данных, протоколов, поддерживать большое число API. PHP поддерживает ООП (деструкторы, открытые, закрытые и защищённые члены и методы, final-члены и методы, интерфейсы и клонирование объектов). PHP поддерживает XML.

2 Методы и средства решения задачи

2.1 Требования к функциональному назначению приложения

Программа должна поддерживать работу разных типов пользователей.

Возможности, которые должны быть реализованы в зависимости от пользователя, представлены в таблице 2.1.

Таблица 2.1 – Возможности пользователей

Тип пользователя	Возможности
Администратор	Формирует списки преподавателей, классов, учащихся. Связывает преподавателя с определенной дисциплиной. Привязывает к классам преподавателей. Просматривает статистику и диаграммы успеваемости.
Преподаватель	Выставляет оценки учащимся, просматривает списки классов, добавляет объявления для класса. Добавляет домашнее задание учащимся.
Ученик (Родитель)	Просматривает успеваемость (выставленные оценки), отправляет и получает сообщения от других учеников класса, просматривает объявления преподавателей, ведущих занятия, а также статистику успеваемости

Доступ к возможностям системы имеют только те пользователи, которые прошли авторизацию посредством кода доступа.

В соответствии с основными функциями были разработаны диаграммы вариантов использования для каждого типа пользователей сайта.

2.2 Анализ требований к интерфейсу приложения

Главное требование к интерфейсу пользователя это его простота. При работе с программой должна быть обеспечена минимальность затрат ресурсов со стороны пользователя. Основная рабочая форма должна включать в себя ссылки на необходимые страницы и рабочую область – область ввода данных. Должна быть предусмотрена возможность перед началом работы с приложением идентификации пользователя системы (проверки кода доступа). Далее должны быть доступны только те элементы интерфейса, которые реализуют функции, соответствующие правам доступа идентифицированного пользователя системы.

Общие требования к пользовательскому интерфейсу можно сформулировать в следующих пунктах:

- разработанный интерфейс должен в наиболее удобной форме отображать все возможности информационной системы,
- должны быть разработаны формы для работы с данными в режиме просмотра и модификации, содержащие соответствующие компоненты для наглядного отображения данных,
- реализовать возможность выставление оценок преподавателями только тем классам, в которых они проводят занятия,
- выбор тех или иных компонентов должен быть обусловлен целесообразностью и удобством для пользователя.

2.2.1 Требования к средствам разработки

Проектирование базы данных предполагается осуществить с использованием технологии IDEF1X.

IDEF1X (IDEF1 Extended) - Data Modeling — методология построения реляционных структур (баз данных), относится к типу методологий «Сущность-взаимосвязь» (ER - Entity-Relationship) и, как правило, используется

для моделирования реляционных баз данных, имеющих отношение к рассматриваемой системе.

Для создания приложения был выбран язык PHP.

PHP (англ. PHP: Hypertext Preprocessor - «PHP: препроцессор гипертекста», англ. Personal Home Page Tools (устар.) - «Инструменты для создания персональных веб-страниц») - скриптовый язык программирования общего назначения, интенсивно применяющийся для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков программирования, применяющихся для создания динамических веб-сайтов. Язык PHP и его интерпретатор разрабатываются группой энтузиастов в рамках проекта с открытым кодом. Проект распространяется под собственной лицензией, несовместимой с GNU GPL.

Для разработки базы данных был использован язык MySQL.

MySQL - свободная система управления базами данных (СУБД). MySQL является собственностью компании Sun Microsystems, осуществляющей разработку и поддержку приложения.

Помимо этого, разработчики создают функциональность по заказу лицензионных пользователей, именно благодаря такому заказу почти в самых ранних версиях появился механизм репликации.

MySQL является решением для малых и средних приложений. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей. Более того, СУБД MySQL поставляется со специальным типом таблиц

EXAMPLE,демонстрирующим принципы создания новых типов таблиц. Благодаряоткрытой архитектуре, в СУБД MySQL постояннопоявляются новые типы таблиц.

Исходя из требований к разрабатываемому приложению, необходимо спроектировать его общую структуру и способ взаимодействия компонент.

2.3 Проектирование программного обеспечения

2.3.1Проектирование структур данных

2.3.1.1 Концептуальное проектирование

Процесс построения инфологической модели состоит из следующих шагов:

- определение сущностей,
- определение зависимостей между сущностями,
- задание первичных и альтернативных ключей,
- определение атрибутов сущностей.

Пользователь, работающий с программными средствами контроля успеваемости учащихся, будет добавлять информацию в базу и работать с ней.

Далее в таблице 2.2 приведен обзор сущностей, где представлены все спроектированные сущности БД, заголовки столбцов означают следующее:

- название сущности – название выделенной в предметной области сущности,
- идентификация таблицы – физическое имя сущности,
- количество записей – предлагаемое количество записей сущности,
- описание – описание сущности.

Таблица 2.2 – Таблица сущностей

Название	Идентификатор	Описание
Ученики	users	Информация о ученике
Школы	schools	Информация о школе
Преподаватели	prepods	Информация о преподавателе
Предметы	predmets	Информация о дисциплинах
Предметы- группы	predmet_groups	Таблица связи преподавателя, группы и предмета
Сообщения	message	Сообщения
Объявления	advert	Объявления
Группы	groups	Информация о классах
Четверть	chetvert	Диапазон дат четверти
Тема	Tema	Тема урока
Задание	zadanie	Домашнее задание
Журнал	journal	Журнал успеваемости

2.3.2. Логическое проектирование

Логическое проектирование включает в себя следующие этапы:

- определение набора отношений исходя из концептуальной модели БД,
- проверка модели с помощью правил нормализации,
- проверка модели в отношении модели пользователей,
- создание полной атрибутивной диаграммы в рамках технологии IDEF1X,
- определение требующейся поддержки целостности.

Анализ функциональных зависимостей в процессе логического проектирования для каждого отношения приведен ниже.

Отношение: Ученики.

Схема отношения: Ученики (идентификатор ученика, идентификатор группы, имя ученика, код доступа ученика).

Первичный ключ: идентификатор ученика.

Альтернативный ключ: имя ученика.

Функциональные зависимости:

Идентификатор ученика -идентификатор группы, имя ученика, код доступа ученика.

Отношение находится в 3НФ.

Отношение: Группа.

Схема отношения: Группа (идентификатор группы, идентификатор школы, название группы, идентификатор преподавателя).

Первичный ключ: идентификатор группы.

Альтернативный ключ: название группы.

Функциональные зависимости:

Идентификатор группы - идентификатор школы, название группы, идентификатор преподавателя.

Отношение находится в 3НФ.

Отношение: Преподаватели.

Схема отношения: Преподаватели (идентификатор преподавателя, идентификатор школы, имя преподавателя, идентификатор предмета, код доступа преподавателя).

Первичный ключ: идентификатор преподавателя.

Альтернативный ключ: имя преподавателя.

Функциональные зависимости:

Идентификатор преподавателя - идентификатор школы, имя преподавателя, идентификатор предмета, код доступа преподавателя.

Отношение находится в 3НФ.

Отношение: Предметы.

Схема отношения: Предметы (идентификатор предмета, название предмета, описание предмета).

Первичный ключ: идентификатор предмета.

Альтернативный ключ: название предмета.

Функциональные зависимости:

Идентификатор предмета- название предмета, описание предмета.

Отношение находится в 3НФ.

Отношение: Школы.

Схема отношения: Школы (идентификатор школы, название школы, адрес школы, телефон школы, имя директора школы, код доступа школы).

Первичный ключ: идентификатор школы.

Альтернативный ключ: название школы.

Функциональные зависимости:

Идентификатор школы- название школы, адрес школы, телефон школы, имя директора школы, код доступа школы.

Отношение находится в 3НФ.

Отношение: Предметы-группы.

Схема отношения: Предметы-группы (идентификатор Предметы-группы, идентификатор группы, идентификатор предмета, идентификатор преподавателя).

Первичный ключ: идентификатор Предметы-группы.

Альтернативный ключ: идентификатор группы, идентификатор предмета, идентификатор преподавателя.

Функциональные зависимости:

Идентификатор Предметы-группы - идентификатор группы, идентификатор предмета, идентификатор преподавателя.

Отношение находится в 3НФ.

Отношение: Сообщения.

Схема отношения: Сообщения (идентификатор сообщения, идентификатор получателя сообщения, идентификатор отправителя сообщения, текст сообщения, дата опрвления сообщения, метка о прочтении сообщения).

Первичный ключ: идентификатор сообщения.

Альтернативный ключ: текст сообщения.

Функциональные зависимости:

Идентификатор сообщения- идентификатор получателя сообщения, идентификатор отправителя сообщения, текст сообщения, дата опрвления сообщения, метка о прочтении сообщения.

Отношение находится в 3НФ.

Отношение: Объявления.

Схема отношения: Объявления (идентификатор объявления, идентификатор отправителя объявления, идентификатор группы получателя объявления, текст объявления, дата отправки объявления, дата окончания срока действия объявления).

Первичный ключ: идентификатор объявления.

Альтернативный ключ: текст объявления.

Функциональные зависимости:

Идентификатор объявления- идентификатор отправителя объявления, идентификатор группы получателя объявления, текст объявления, дата отправки объявления, дата окончания срока действия объявления.

Отношение находится в 3НФ.

Отношение: Журнал.

Схема отношения: Журнал (идентификатор оценки, дата выставления оценки, идентификатор предмет-группа, идентификатор ученика, оценка).

Первичный ключ: идентификатор объявления.

Альтернативный ключ: идентификатор ученика.

Функциональные зависимости:

Идентификатор оценки- дата выставления оценки, идентификатор предмет-группа, идентификатор ученика, оценка.

Отношение находится в 3НФ.

Отношение: Тема.

Схема отношения: Тема (идентификатор темы, идентификатор предмет-группа,название темы, дата урока, дополнительная информация).

Первичный ключ: Идентификатор темы.

Альтернативный ключ: идентификатор предмет-группа.

Функциональные зависимости:

Идентификатор темы- идентификатор предмет-группа,название темы, дата урока, дополнительная информация.

Отношение находится в 3НФ.

Отношение: Задание.

Схема отношения: Задание (идентификатор задания, идентификатор предмет-группа, задание, дата задания).

Первичный ключ: Идентификатор задания.

Альтернативный ключ: идентификатор предмет-группа.

Функциональные зависимости:

Идентификатор задания- идентификатор предмет-группа, задание, дата задания.

Отношение находится в 3НФ.

Отношение: Четверть.

Схема отношения: Четверть (идентификатор четверти, идентификатор школы, название четверти, дата начала четверти, дата конца четверти).

Первичный ключ: Идентификатор четверти.

Альтернативный ключ: идентификатор школы.

Функциональные зависимости:

Идентификатор четверти- идентификатор школы,название четверти, дата начала четверти, дата конца четверти

Отношение находится в 3НФ.

2.3.3 Даталогическое проектирование

На данном этапе разработки БД была сформирована физическая модель БД, изображенная в таблицах 2.3 -2.14.

Таблица 2.3 – Отношение «Объявления» («advert»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
Advert_id	Идентификатор объявления	int	11	-	PrimaryKey
Out_id	Идентификатор отправителя	int	11	-	ForeignKey
In_id	Идентификатор получателя	int	11	-	ForeignKey
advert	Объявление	Varchar	255	-	Attribute
Advert_date	Дата добавления объявления	datetime			Attribute
Advert_out_date	Дата актуальности объявления	Datetime			Attribute

Таблица 2.4 – Отношение «Группа» («groups»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
Group_id	Идентификатор группы	int	11	-	PrimaryKey
School_id	Идентификатор школы	int	11	-	ForeignKey
Group_name	Название группы	varchar	255	-	Attribute
Prepod_id	Идентификатор преподавателя	int	11	-	ForeignKey

Таблица 2.5 – Отношение «Журнал» («Journal»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
Journal_id	Идентификатор группы	int	11	-	ForeignKey
data	Идентификатор школы	date		-	Attribute
Predmet_group_id	Название группы	Int	11	-	ForeignKey
User_id	Идентификатор преподавателя	int	11	-	ForeignKey
ocenka	Оценка	int	11		Attribute

Таблица 2.6 – Отношение «Сообщение» («message»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
Message_id	Идентификатор сообщения	int	11	-	PrimaryKey
Out_id	Идентификатор отправителя	int	11	-	ForeignKey
In_id	Идентификатор получателя	Int	11	-	ForeignKey
message	Сообщение	Varchar	255	-	Attribute
Message_date	Дата отправки сообщения	datetime			Attribute
select	Маркер сообщения	int	1	-	Attribute

Таблица 2.7 – Отношение «Предмет-группа» («predmet_groups»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
Predmet_group_id	Идентификаторпредмет группа	int	11	-	PrimaryKey
Group_id	Идентификатор группы	int	11	-	ForeignKey
Predmet_id	Идентификатор предмета	Int	11	-	ForeignKey
Prepod_id	Идентификатор преподавателя	int	11	-	ForeignKey

Таблица 2.8 – Отношение «Предметы» («predmets»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
Predmet_id	Идентификаторпредмета	int	11	-	PrimaryKey
Predmet_name	Название предмета	Varchar	255	-	Attribute
Predmet_text	Описание предмета	Varchar	255	-	Attribute

Таблица 2.9 – Отношение «Преподаватели» («prepods»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
Prepod_id	Идентификатор преподавателя	int	11	-	PrimaryKey
School_id	Идентификатор школы	int	11	-	ForeignKey
Prepod_name	Имя преподавателя	Varchar	255	-	Attribute
Predmet_id	Идентификатор предмета	Int	11	-	ForeignKey
Prepod_psw	Код доступа преподавателя	Varchar	255	-	Attribute

Таблица 2.10 – Отношение «Школа» («schools»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
school_id	Идентификатор школы	int	11	-	PrimaryKey
school_name	Название школы	Varchar	255	-	Attribute
School_adress	Адрес школы	Varchar	255	-	Attribute
School_tel	Телефон школы	Varchar	255	-	Attribute
School_dir_name	ФИО директора	Varchar	255	-	Attribute
School_photo	Фотография школы	Varchar	255	null	Attribute
School_psw	Код доступа	Varchar	255	-	Attribute

Таблица 2.11 – Отношение «Ученики» («users»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
user_id	Идентификатор ученика	int	11	-	PrimaryKey
Group_id	Идентификатор группы	Varchar	255	-	ForeignKey
User_name	Имя ученика	Varchar	255	-	Attribute
User_psw	Код доступа	Varchar	255	-	Attribute

Таблица 2.12 – Отношение «Тема» («tema»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
tema_id	Идентификатор темы	int	11	-	PrimaryKey
Predmet_group_id	Идентификатор предмет-	int	255	-	ForeignKey

	группа				
Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
tema	Название темы	Varchar	255	-	Attribute
Tema_date	Дата темы	date		-	Attribute
Dop_info	Дополнительная инф.	Varchar	255	-	Attribute

Таблица 2.13 – Отношение «Задание» («zadanie»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
zadanie_id	Идентификатор темы	int	11	-	PrimaryKey
Predmet_group_id	Идентификатор предмет-группа	int	255	-	ForeignKey
Zadanie	Задание	Varchar	255	-	Attribute
zadanie_date	Дата задания	date		-	Attribute

Таблица 2.14 – Отношение «Четверть» («chetvert»)

Имя столбца	Содержательное описание	Тип данных	Размерность	Возможность значения Null	Роль
chetvert_id	Идентификатор темы	int	11	-	PrimaryKey
School_id	Идентификатор школы	Int	11	-	ForeignKey
Chetvert_name	Идентификатор предмет-группа	char	255	-	Attribute
Chetvert_begin	Дата начала четверти	date		-	Attribute
Chetvert_end	Дата окончания четверти	date		-	Attribute

2.3.4Инфологическая модель

В базах данных содержится информация об объектах, существующих в реальном мире. Все эти объекты являются материальными, обладающим некоторыми свойствами и присущим им поведением. При создании программного продукта, свойства объектов хранятся в базе данных, а их поведение реализуется в приложении посредством методов - процедур, которые представляют собой отклик на события инициируемых пользователем.

Инфологическая модель является проблемно-ориентированной и системно-независимой, т.е. не зависимой от конкретной СУБД, операционной системы и аппаратного обеспечения ЭВМ.

Основным требованием к инфологической модели, вытекающим из ее назначения, является требование адекватного отображения предметной области. Инфологическая модель должна быть непротиворечивой. Она является единым интегрированным описанием предметной области и отражает взгляды и потребности всех пользователей системы

Инфологическая модель представляет информационные потоки, сущности и связи данной предметной области. Она может быть представлена в виде ER-модели и реляционной схемы.

Целью построения этой структуры является выявление и объединение информационных требований пользователя, связей между элементами данных без относительно к их содержанию и среде их хранения. Инфологическая модель должна обладать свойством легкой расширяемости, обеспечивающим ввод новых данных без изменений ранее определенных.

Центральной компонентой инфологической модели является описание объектов предметной области и связей между ними (ER-модель).

Цель инфологического моделирования - обеспечение наиболее естественных для человека способов сбора и представления той информации, которую предполагается хранить в создаваемой базе данных. Поэтому инфологическую модель данных пытаются строить по аналогии с естественным

языком (последний не может быть использован в чистом виде из-за сложности компьютерной обработки текстов и неоднозначности любого естественного языка).

Основными конструктивными элементами инфологических моделей являются сущности, связи между ними, идентификаторы (ключи) и свойства (атрибуты).

Сущность - любой различимый объект (объект, который мы можем отличить от другого), информацию о котором необходимо хранить в базе данных. Сущностями могут быть люди, места, самолеты, рейсы, вкус, цвет и т.д. Необходимо различать такие понятия, как тип сущности и экземпляр сущности. Понятие тип сущности относится к набору однородных личностей, предметов, событий или идей, выступающих как целое.

Инфологическая модель предметной области приведена на рисунке 2.1.

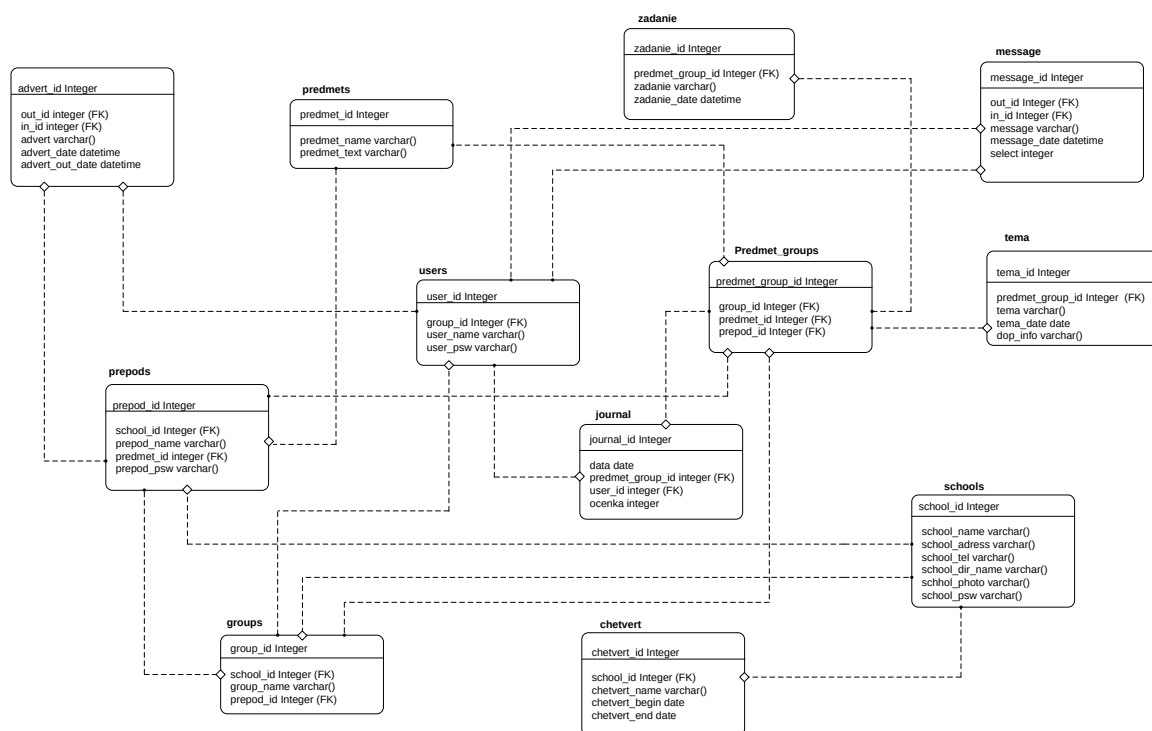


Рисунок 2.1 – Инфологическая модель базы данных

2.4 Проектирование структуры программного обеспечения

При проектировании программных средств системы контроля успеваемости учащихся, для определения архитектуры разрабатываемой системы приведем диаграмму компонентов, изображенную на рисунке 2.2.

Диаграмма компонентов, описывает особенности физического представления системы. Диаграмма устанавливает зависимости между программными компонентами, в роли которых выступают файлы системы. Пунктирные стрелки, соединяющие модули, показывают отношения взаимозависимости. Основными графическими элементами диаграммы компонентов являются компоненты и зависимости между ними.

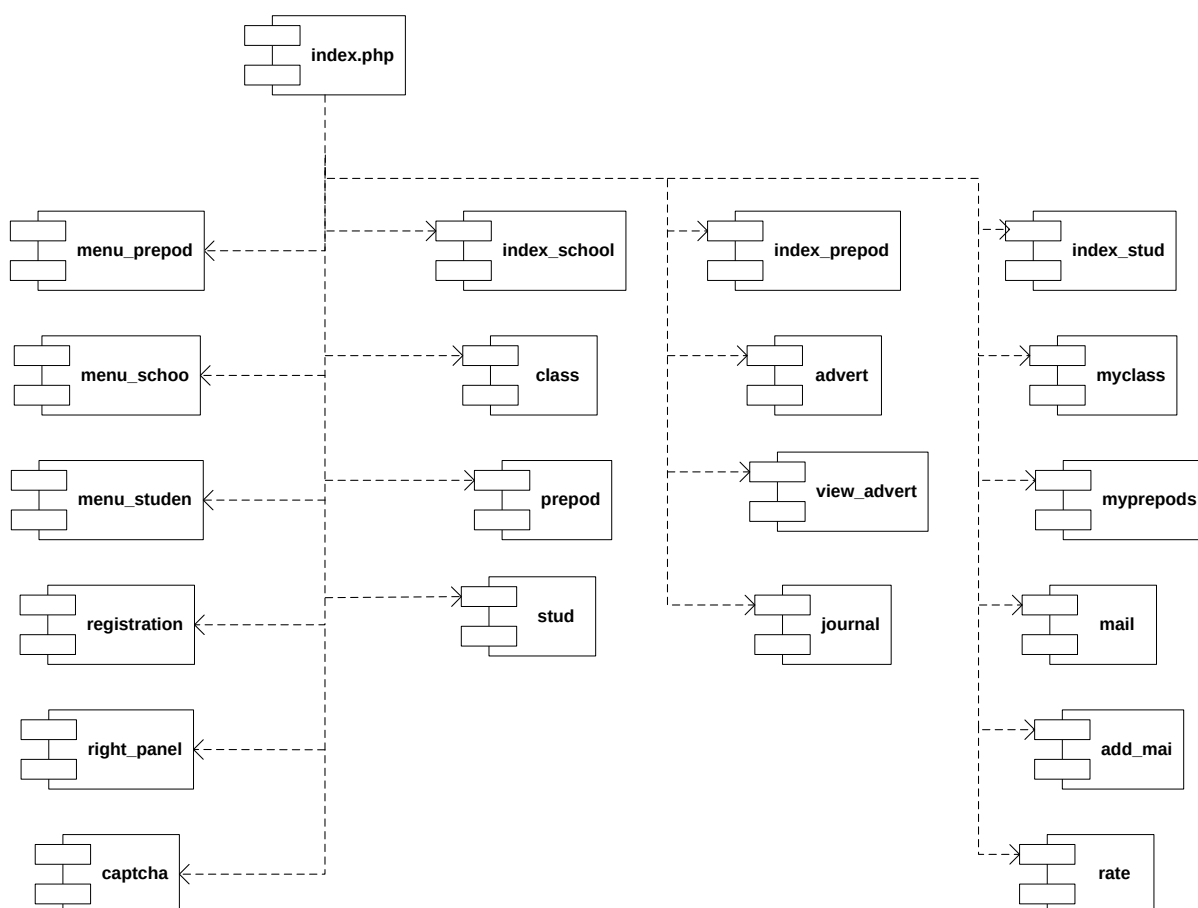


Рисунок 2.2 – Диаграмма компонентов системы

На диаграмме изображено взаимодействие компонентов статистической и пользовательских частей.

Для отображения механизма взаимодействия подсистем, была построена диаграмма коммуникаций, приведенная на рисунке 2.3.

Рассмотрим диаграмму коммуникаций. На ней изображены объекты, которые участвуют в работе системы. Перечислим объекты, присутствующие на диаграмме коммуникаций:

- пользователь (ученик),
- система обработки данных.

Пользователь (ученик) взаимодействует с системой, вводя данные в виде имени предмета. В ответ от системы обработки данных он получает статистические данные в виде количества выставленных оценок, среднего бала и график успеваемости учащихся в его классе.

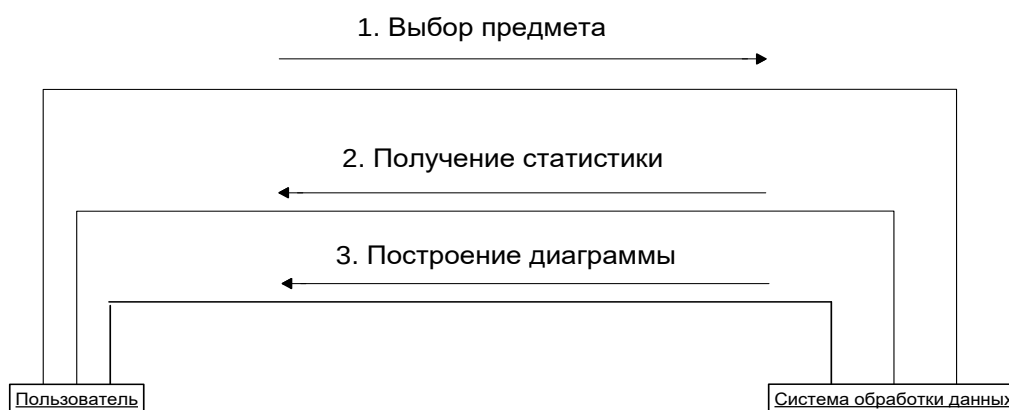


Рисунок 2.3 – Диаграмма коммуникаций для запроса статистики

Для описания взаимодействия преподавателя с системой приведем диаграмму последовательности, изображенную на рисунке 2.4.

На диаграмме последовательности объект “Преподаватель” выбирает название класса из списка, объект “Система” формирует форму для выбора даты выставления оценки и отображает ее на экран.

Объект «Преподаватель» выбирает с помощью формы дату выставления оценки. «Система» запрашивает у «СУБД» список класса и формирует форму для занесения оценок учащимся.

Объект «Преподаватель» заносит оценки в форму и сохраняет результат. «Система» обрабатывает полученные результаты из формы и заносит их в

электронный журнал (БД), после чего формирует на экране сообщение об удачном выполнении операции.

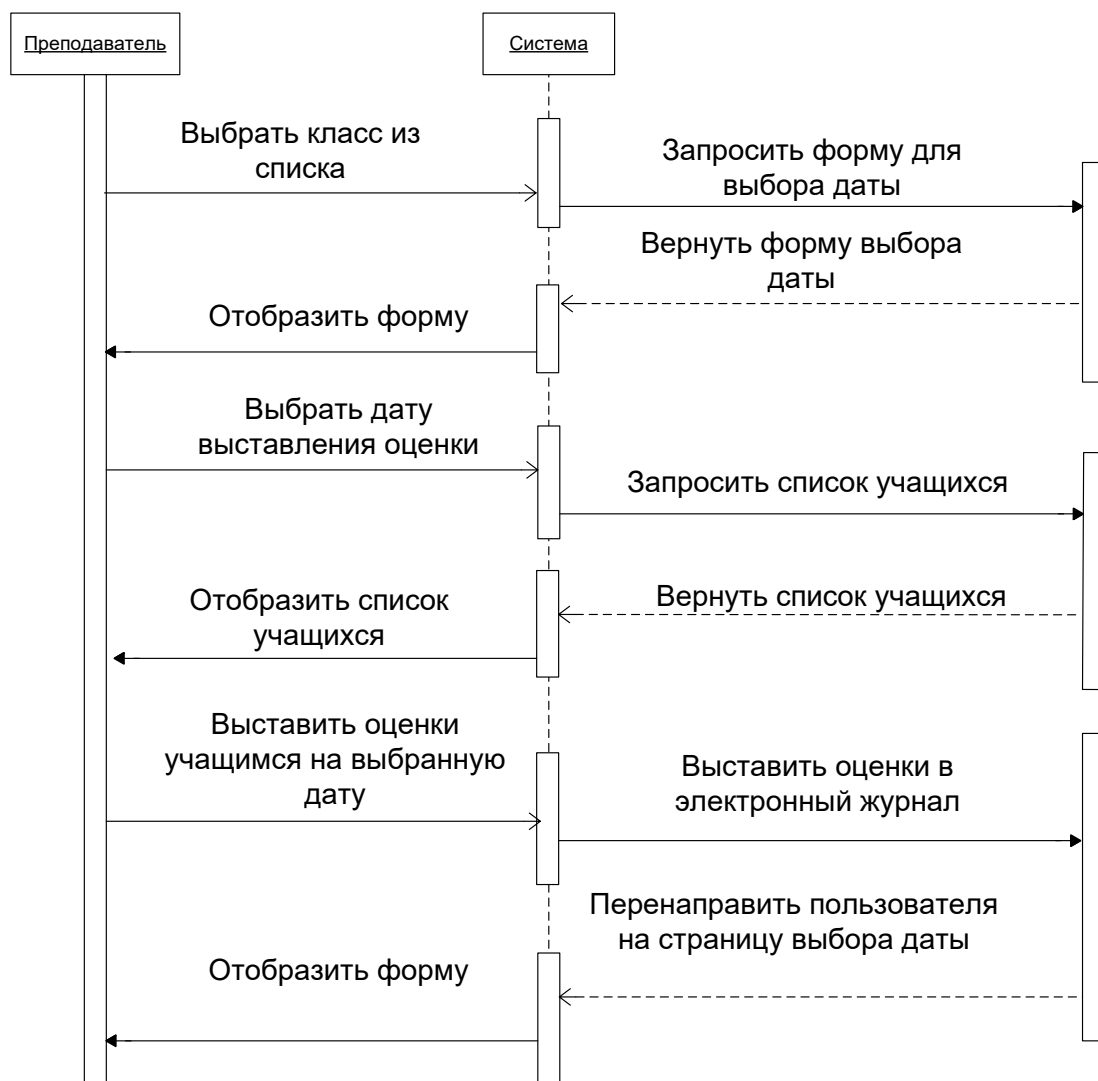


Рисунок 2.4 – Диаграмма последовательности

2.4.1 Проектирование алгоритмов

Рассмотрим алгоритм построения графика успеваемости, который изображен на диаграмме деятельности на рисунке 2.5.

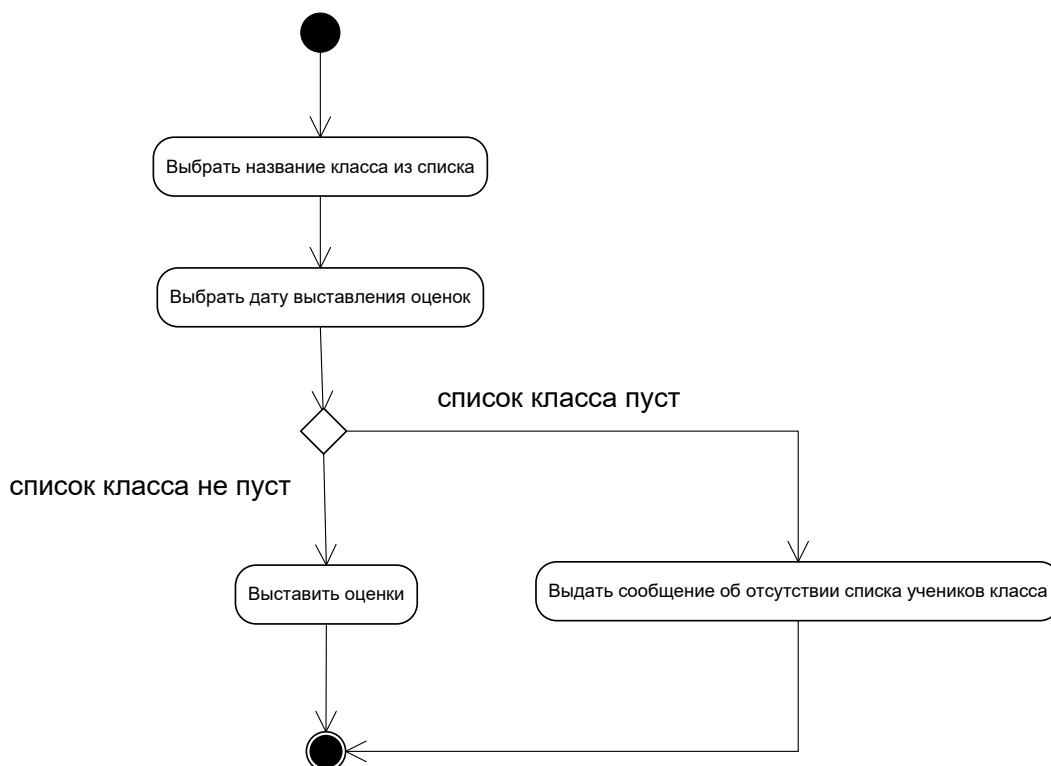


Рисунок 2.5 – Диаграмма деятельности построения графика успеваемости

2.5 Проектирование пользовательского интерфейса

Интерфейс - в широком смысле определенная стандартами граница между взаимодействующими независимыми объектами. Интерфейс задает параметры, процедуры и характеристики взаимодействия объектов.

Интерфейс пользователя - элементы и компоненты программы, которые способны оказывать влияние на взаимодействие пользователя с программным обеспечением.

Пользовательский интерфейс (ПИ) программного средства реализован в соответствии с требованиями к пользовательскому интерфейсу, определенными в процессе анализа требований. Разработан состав, иерархия диалоговых страниц, т.е. логика работы пользовательского интерфейса.

Разработка проводилась на языке РНР.

Далее приводится описание страниц и их элементов с указанием типа и назначения используемых компонент (смотри, соответственно, рисунки 2.6-2.24 и таблицы 2.15-2.33).

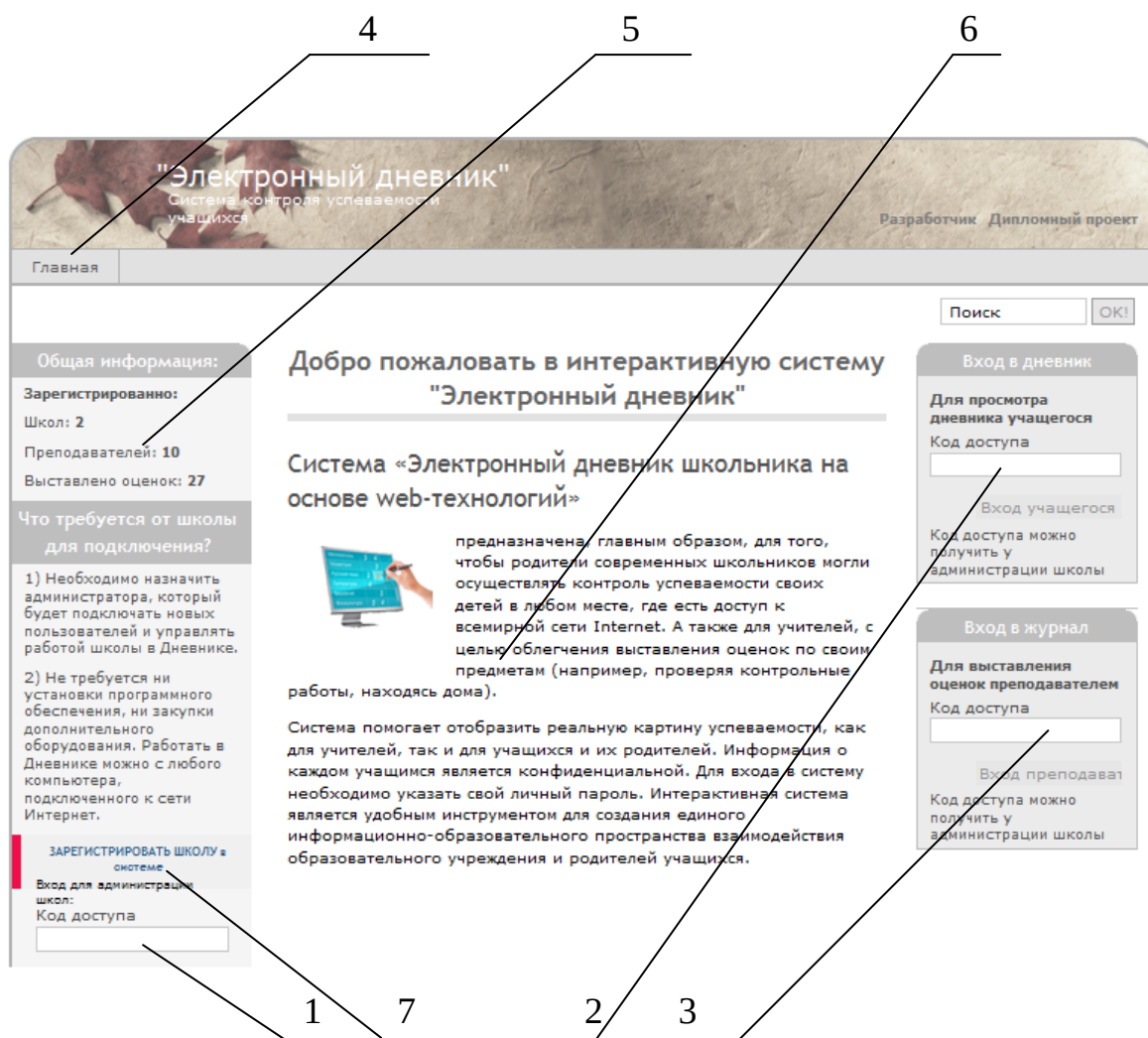


Рисунок 2.6 – Главная страница

Элементы главной страницы показаны в таблице 2.15

Таблица 2.15 -Элементы главной страницы

Обозначение элемента	Компонент	Описание
1	Input type	Поле ввода кода доступа к
2	Input type	Поле ввода кода доступа для учащегося
3	Input type	Поле ввода кода доступа для преподавателя
4	Href	Ссылка на главную страницу
5	Table	Информация о системе
6	Table	Информационное поле
7	Href	Ссылка на окно регистрации школы в системе

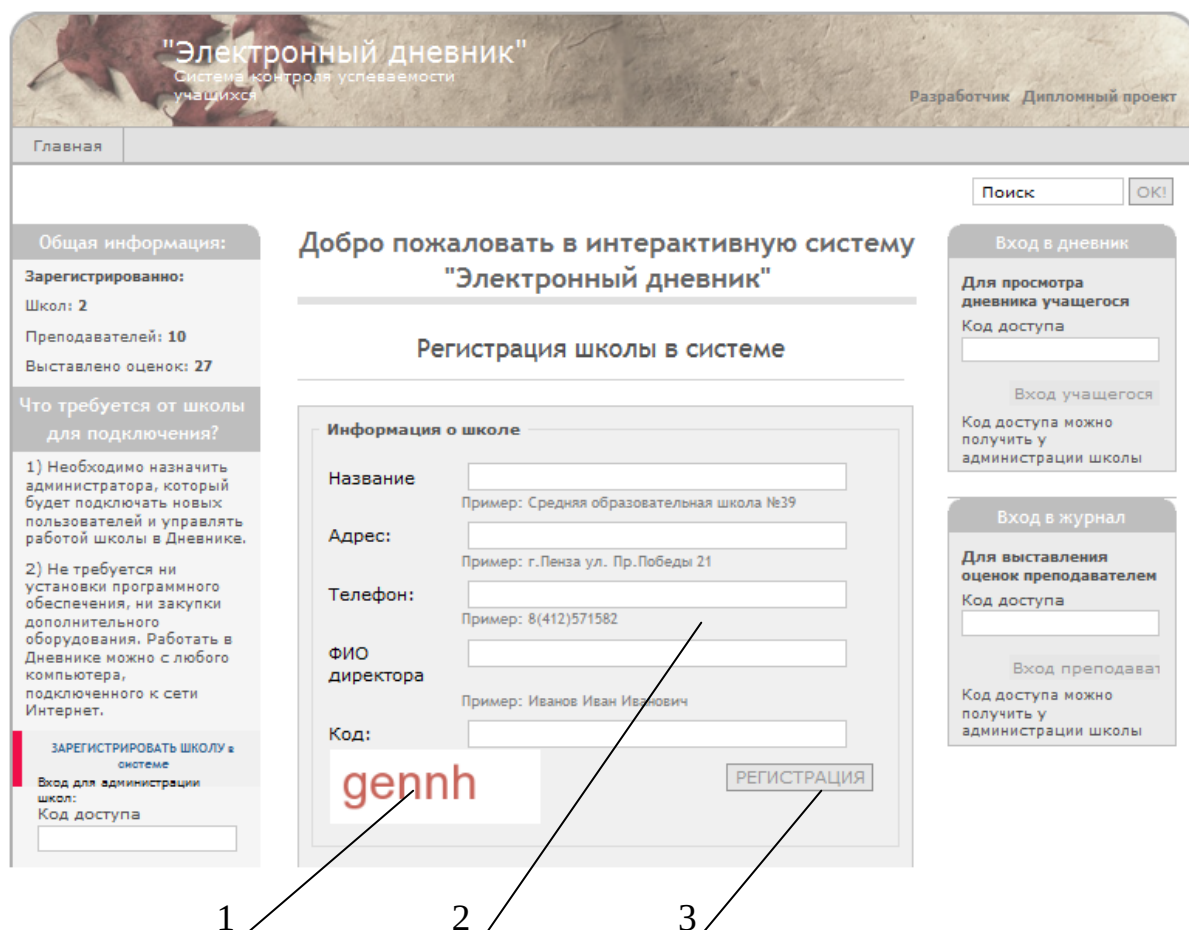


Рисунок 2.7 – Регистрация в системе

Таблица 2.16– Элементы окна регистрации школы

Обозначение элемента	Компонент	Описание
1	img	Изображение кода для подтверждения регистрации
2	form	Форма регистрации
3	button	Кнопка регистрации

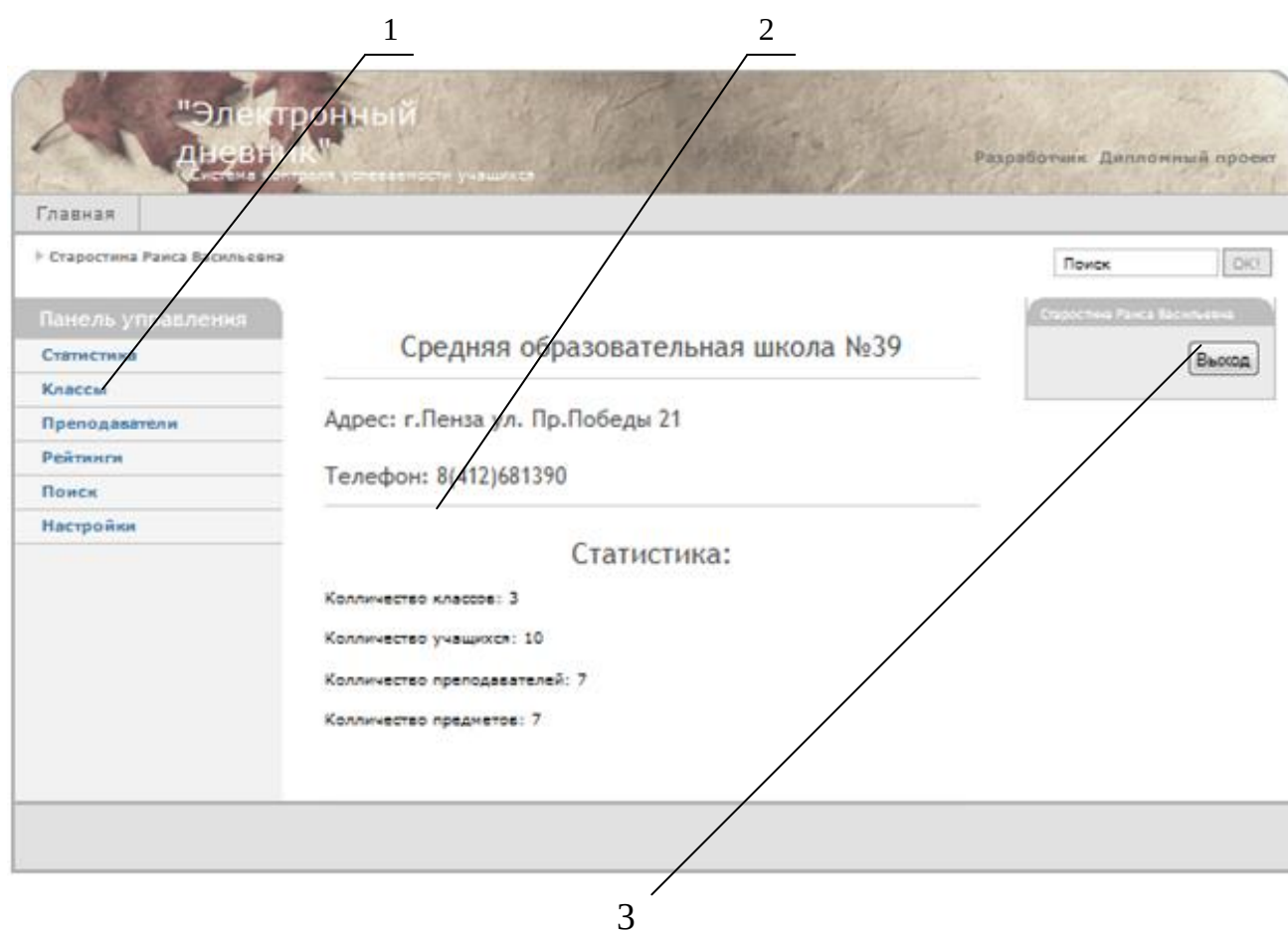


Рисунок 2.8 – Окно администрирования школы

Таблица 2.17 – Элементы окна администрирования школы

Обозначение элемента	Компонент	Описание
1	href	Меню панели управления
2	Table	Информационное поле
3	Form	Кнопка выхода из системы

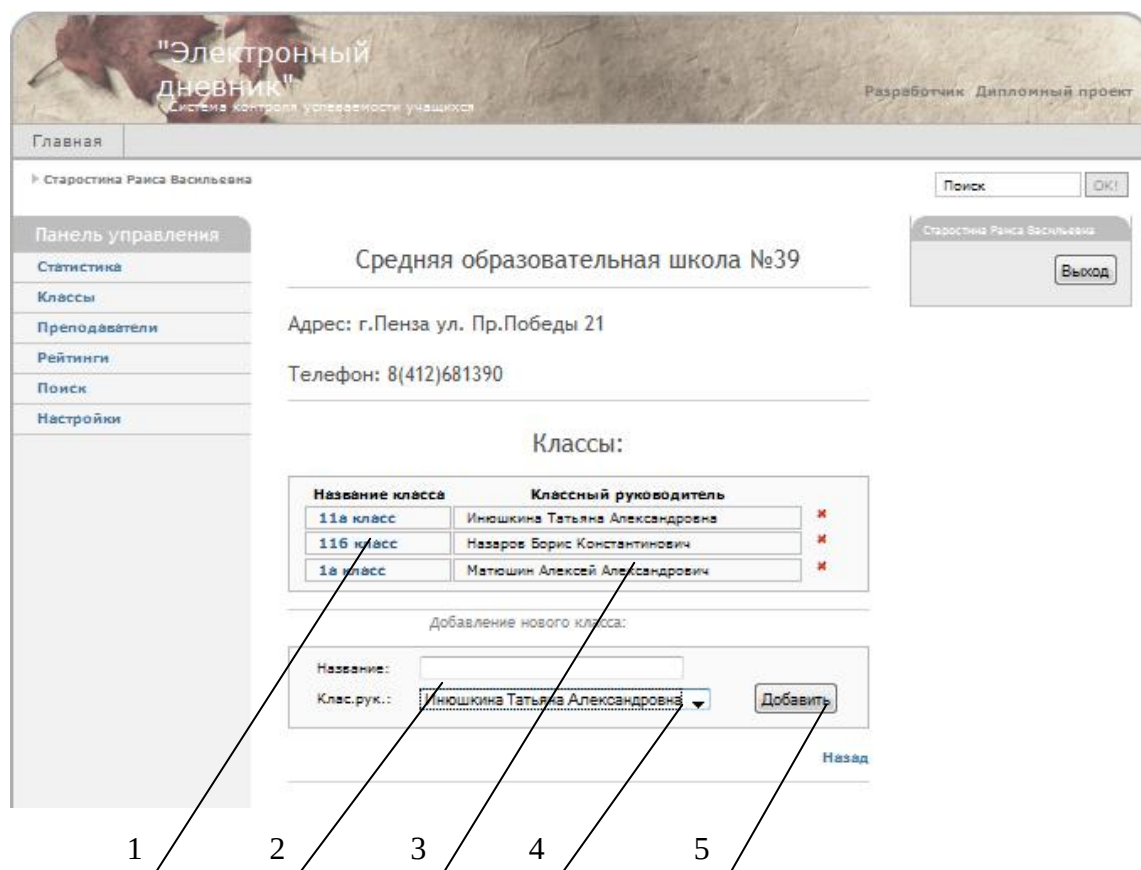


Рисунок 2.9 – Окно «Классы»

Таблица 2.18 – Элементы окна «Классы»

Обозначение элемента	Компонент	Описание
1	href	Ссылка выбора классов
2	Inputtype	Поле ввода нового класса
3	Table	Информация о классном руководителе класса
4	combobox	Выбор классного руководителя из списка преподавателей школы
5	Button	Кнопка добавления нового класса

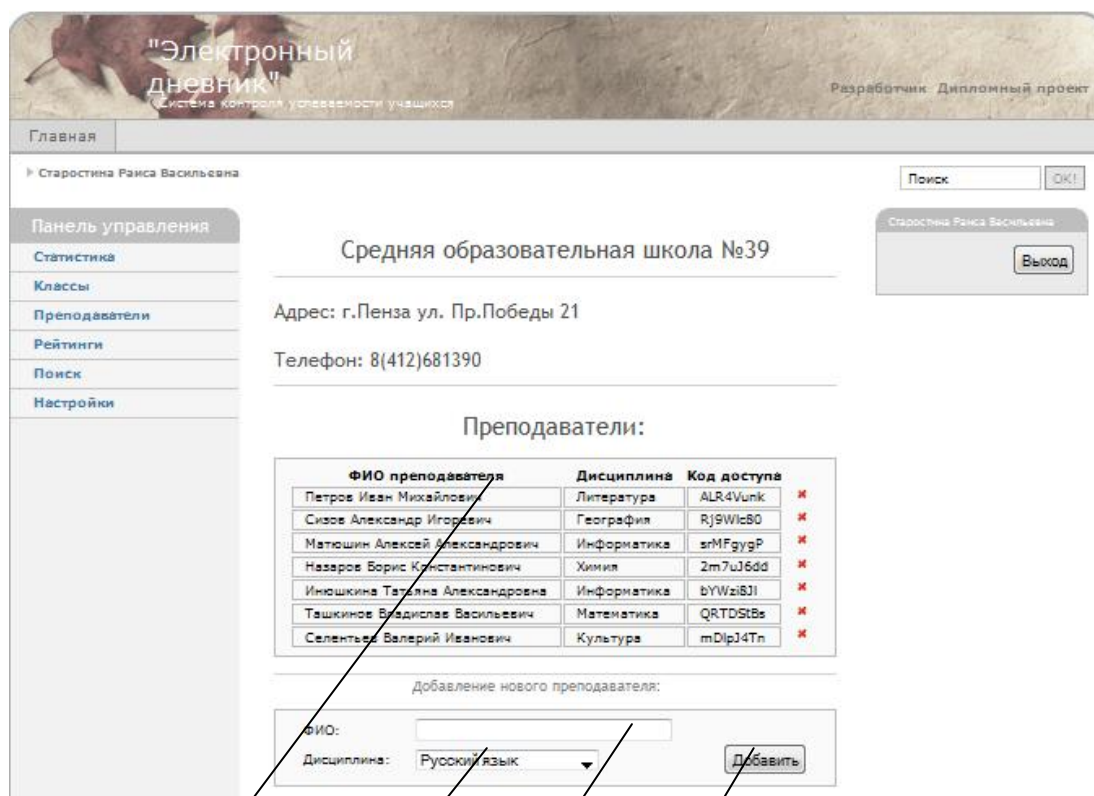


Рисунок 2.10 – Окно «Преподаватели»

Таблица 2.19 – Элементы окна «Преподаватели»

Обозначение элемента	Компонент	Описание
1	Table	Информация о преподавателях
2	combobox	Выбор дисциплины
3	Inputtype	Поле ввода нового преподавателя
4	Button	Кнопка добавления нового преподавателя

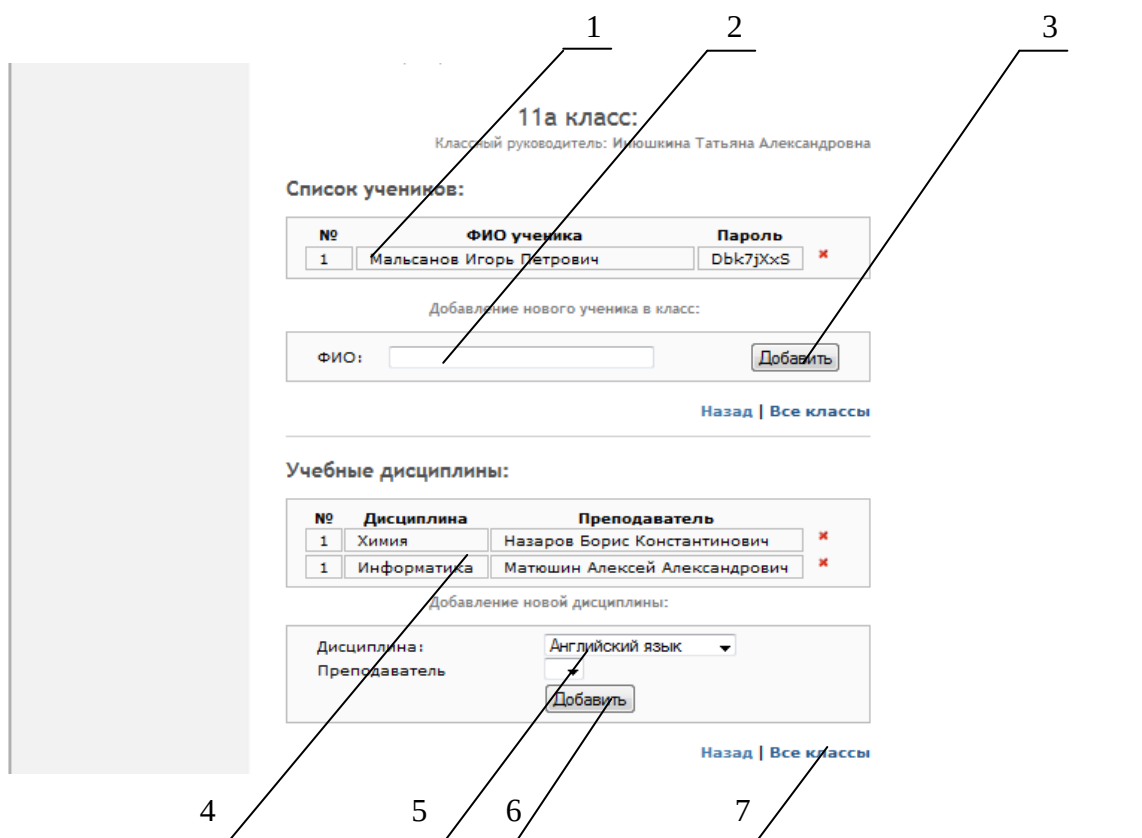


Рисунок 2.11 – Окно «Ученики»

Таблица 2.20– Элементы окна «Ученики»

Обозначение элемента	Компонент	Описание
1	Table	Список учеников
2	Inputtype	Поле ввода нового ученика в класс
3	button	Кнопка добавления нового ученика
4	Table	Список учебных дисциплин класса
5	Combobox	Выбор учебной дисциплины
6	Button	Кнопка добавления новой дисциплины классу
7	combobox	Выбор преподавателя в зависимости от дисциплины

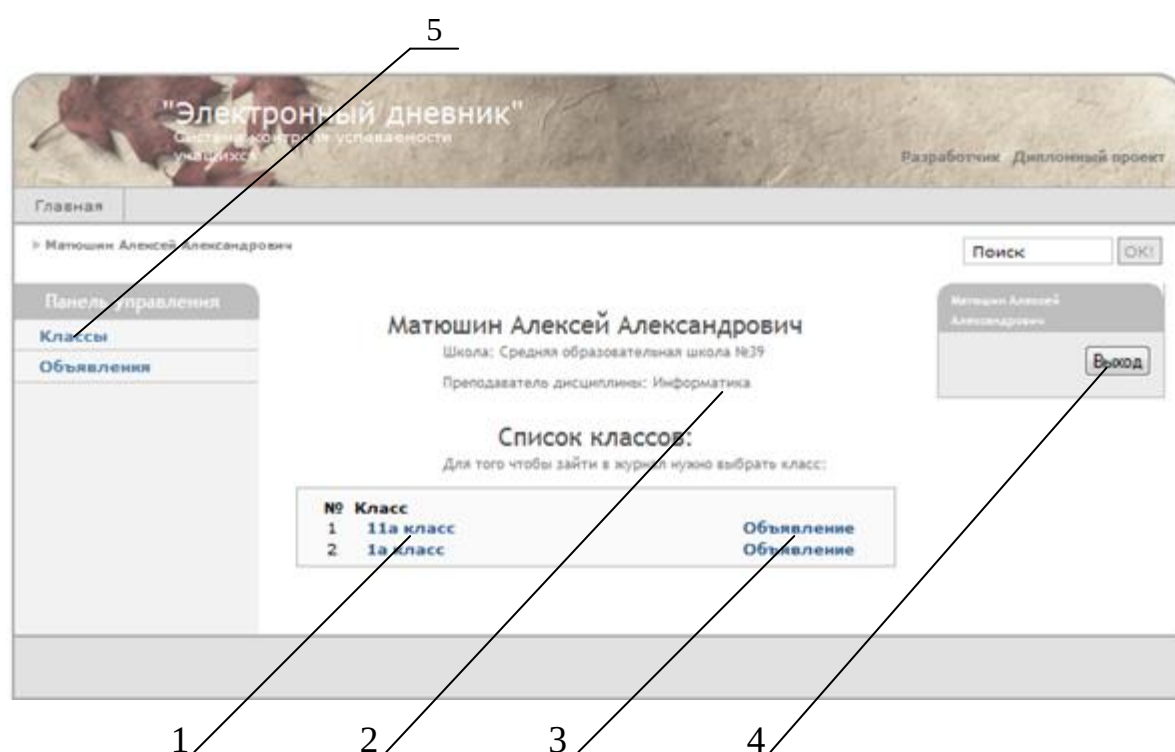


Рисунок 2.12 – Главное окно преподавателя

Таблица 2.21– Элементы главного окна преподавателя

Обозначение элемента	Компонент	Описание
1	Href	Ссылка на просмотр класса
2	Table	Информационная часть
3	Href	Ссылка на добавление объявлений классу
4	Button	Кнопка выхода из системы
5	Href	Меню панели управления

		преподавателя
--	--	---------------

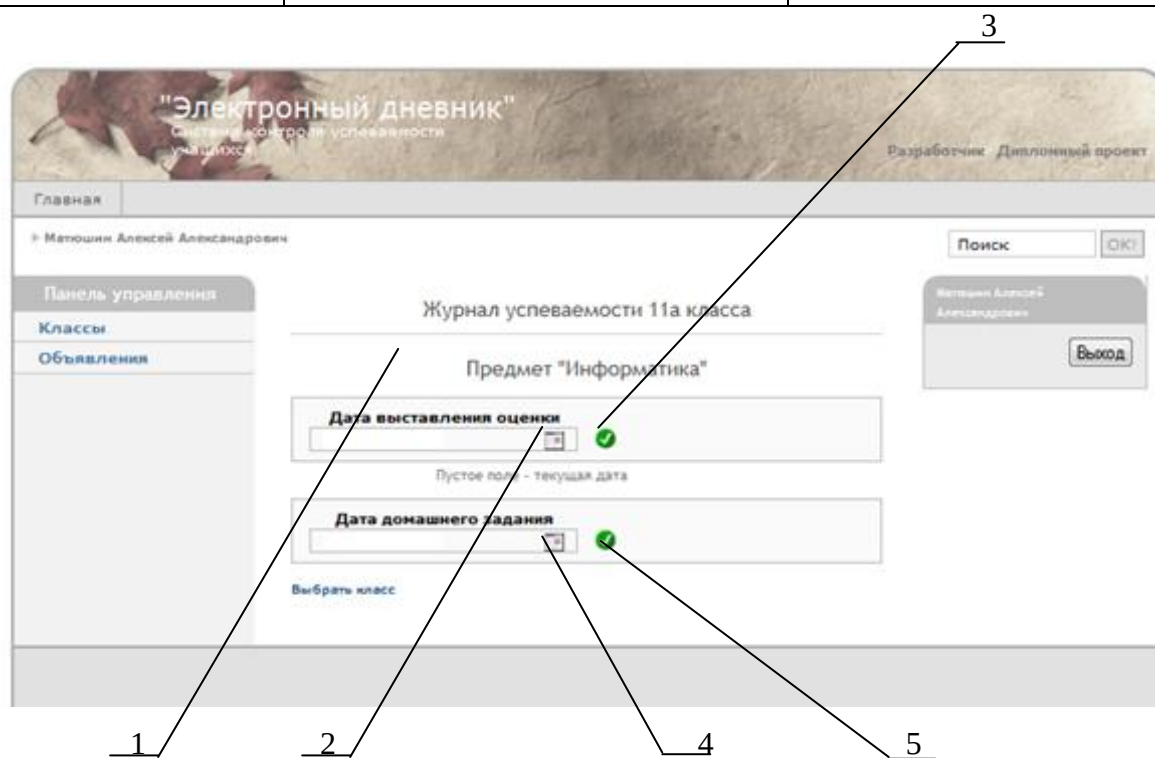


Рисунок 2.13 – Окно выбора меню «Классы»

Таблица 2.22– Элементы окна выбора меню «Классы»

Обозначение элемента	Компонент	Описание
1	Table	Информационное поле
2	Javascript	Выбор даты выставления оценки
3	button	Кнопка выбора даты
4	Javascript	Выбор даты выставления задания
5	button	Кнопка выбора даты

Панель управления

Классы

Объявления

Журнал успеваемости 1а класса

Предмет "Информатика"

Дата выставления оценки

Пустое поле - текущая дата

Тема занятия:

Введите тему занятия

Сохранить

Список класса:

Выставление оценок на 02.12.2010г.

#	ФИО	
1	Родионов Антон Александрович	
2	Иванов Артем Сергеевич	
3	Мельникова Ирина Владимировна	
4	Лобанов Дмитрий Васильевич	
5	Сафимов Марат Рафаэльевич	
6	Сафимов Дамир Михайлович	

Сохранить

Выбрать класс

Матюшин Алексей Александрович

Выход

Рисунок 2.14 – Окно выставления оценок

Таблица 2.23– Элементы окна выставления оценок

Обозначение элемента	Компонент	Описание
1	textarea	Поле ввода темы занятия
2	Button	Кнопка сохранения выставленной оценки
3	Table	Список учащихся
4	Input type	Поле ввода оценки

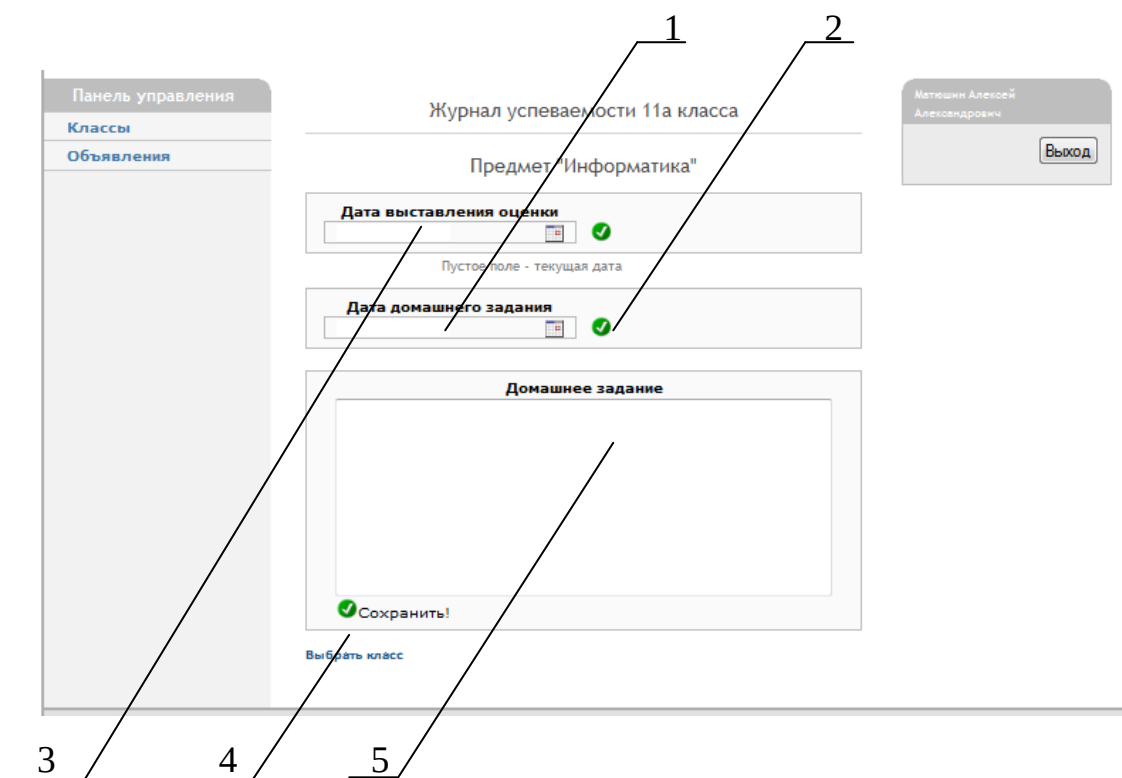


Рисунок 2.15 – Окно выставления оценок

Таблица 2.24 – Элементы окна выставления оценок

Обозначение элемента	Компонент	Описание
1	javascript	Поле выбора даты домашнего задания
2	Button	Кнопка выбора выставленной оценки
3	javascript	Поле выбора даты выставления оценки
4	Button	Кнопка сохранения
5	textarea	Поле ввода домашнего задания

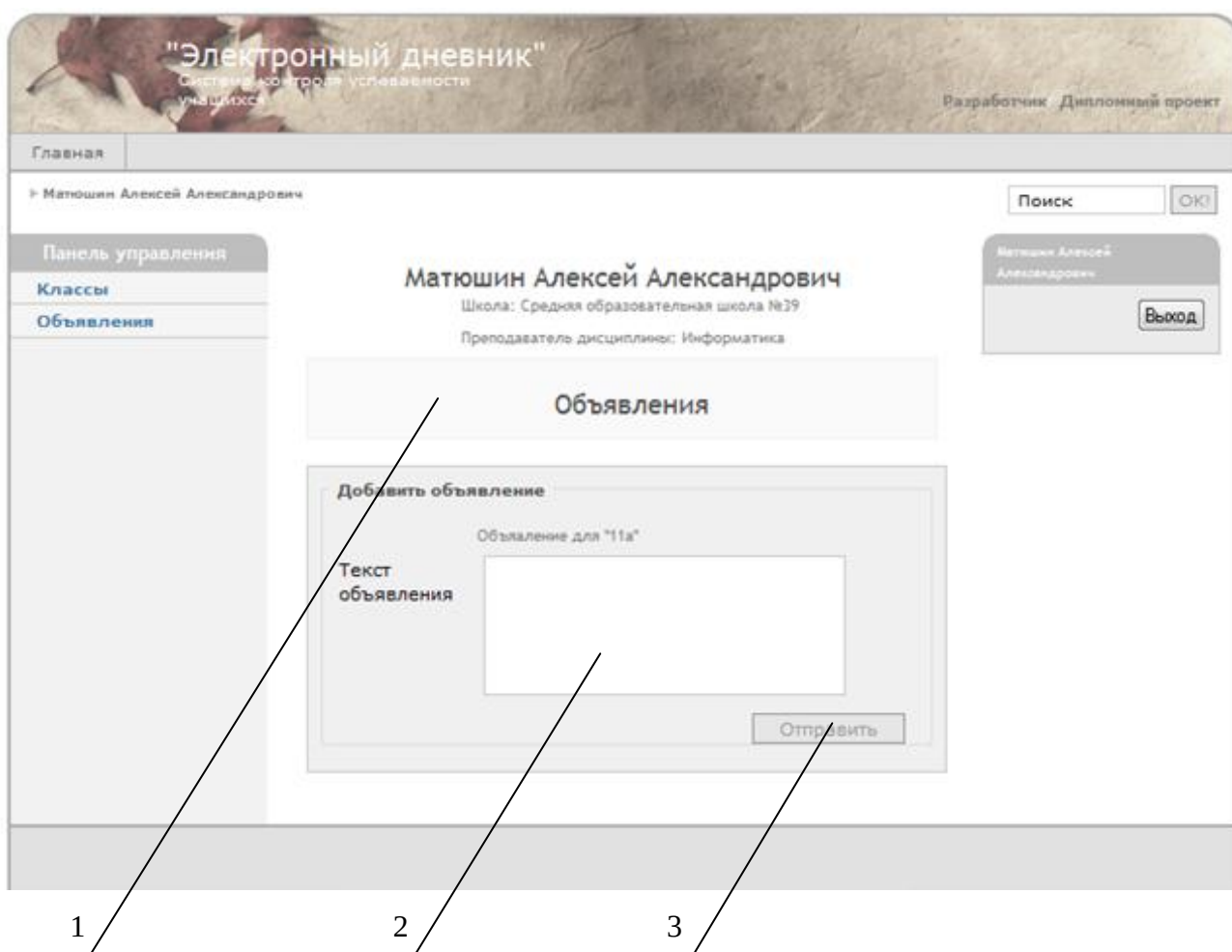


Рисунок 2.16 – Окно добавления объявления

Таблица 2.25– Элементы окна добавления объявления

Обозначение элемента	Компонент	Описание
1	Table	Информационное поле
2	textarea	Поле ввода информации
3	Button	Кнопка добавления объявления

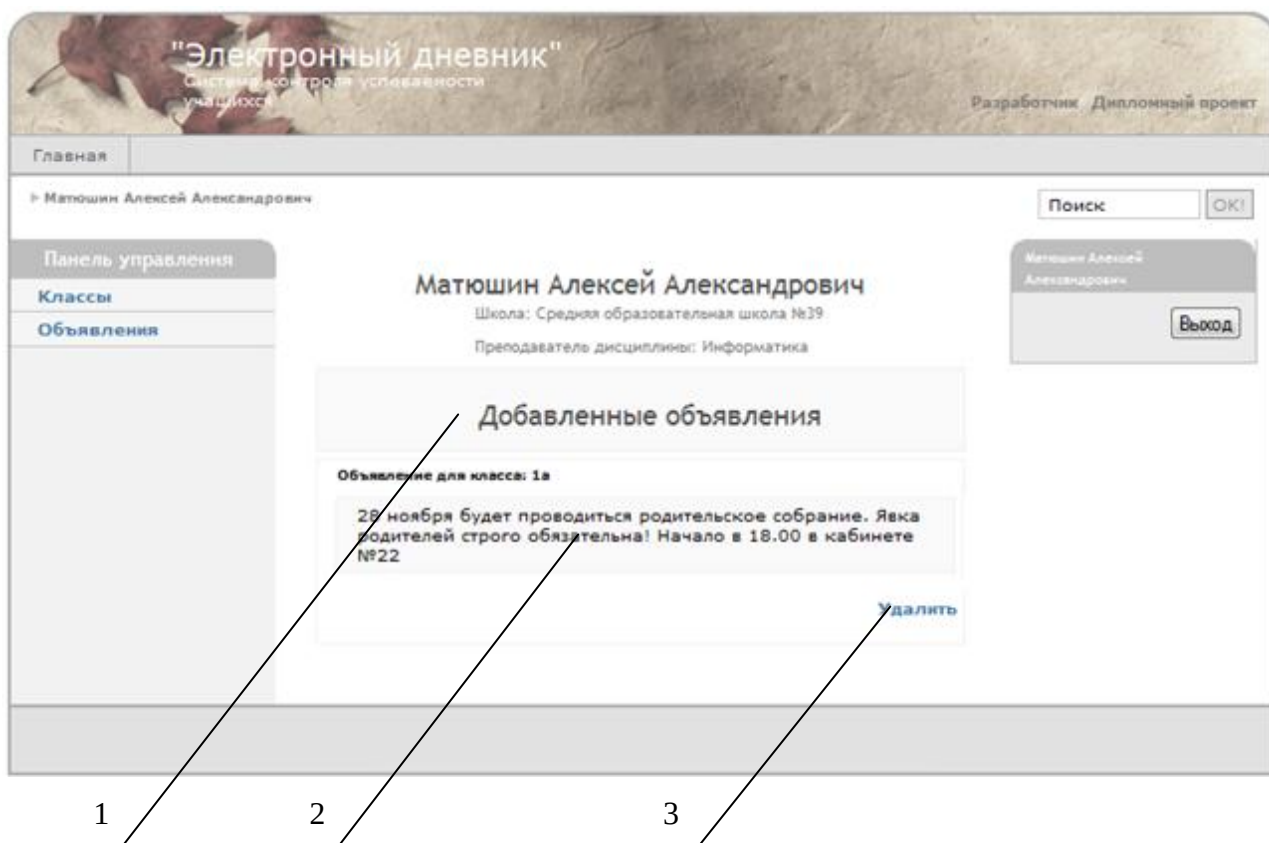


Рисунок 2.17 – Окно меню «Объявления»

Таблица 2.26– Элементы окна меню «Объявления»

Обозначение элемента	Компонент	Описание
1	Table	Информационное поле
2	Table	Список добавленных объявлений
3	Button	Кнопка удаления объявлений

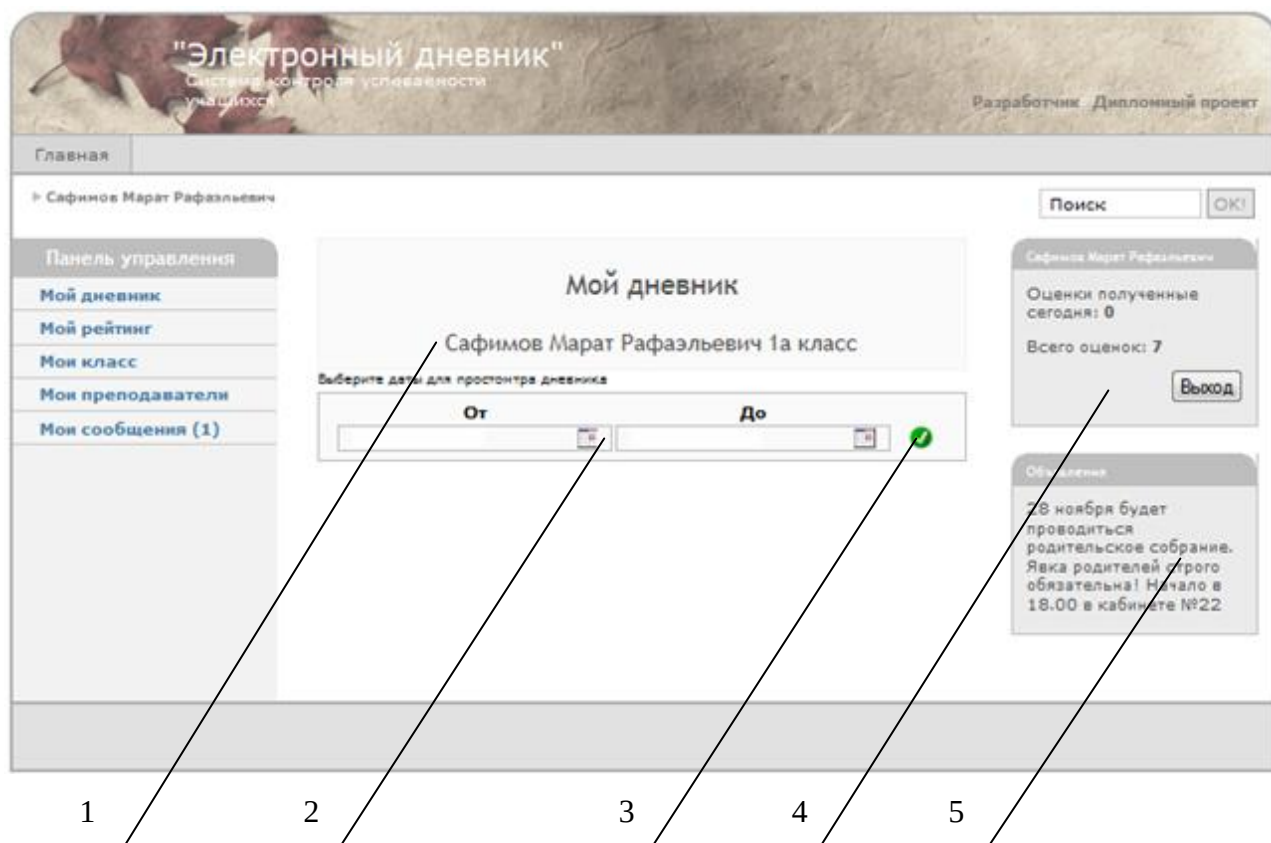


Рисунок 2.18 – Окно учащегося «Мой дневник»

Таблица 2.27– Элементы окна учащегося «Мой дневник»

Обозначение элемента	Компонент	Описание
1	Table	Информационное поле
2	javascript	Выбор диапазона чисел для просмотра дневника
3	Button	Кнопка просмотра дневника
4	Table	Статистика количества полученных оценок
5	Table	Окно объявлений

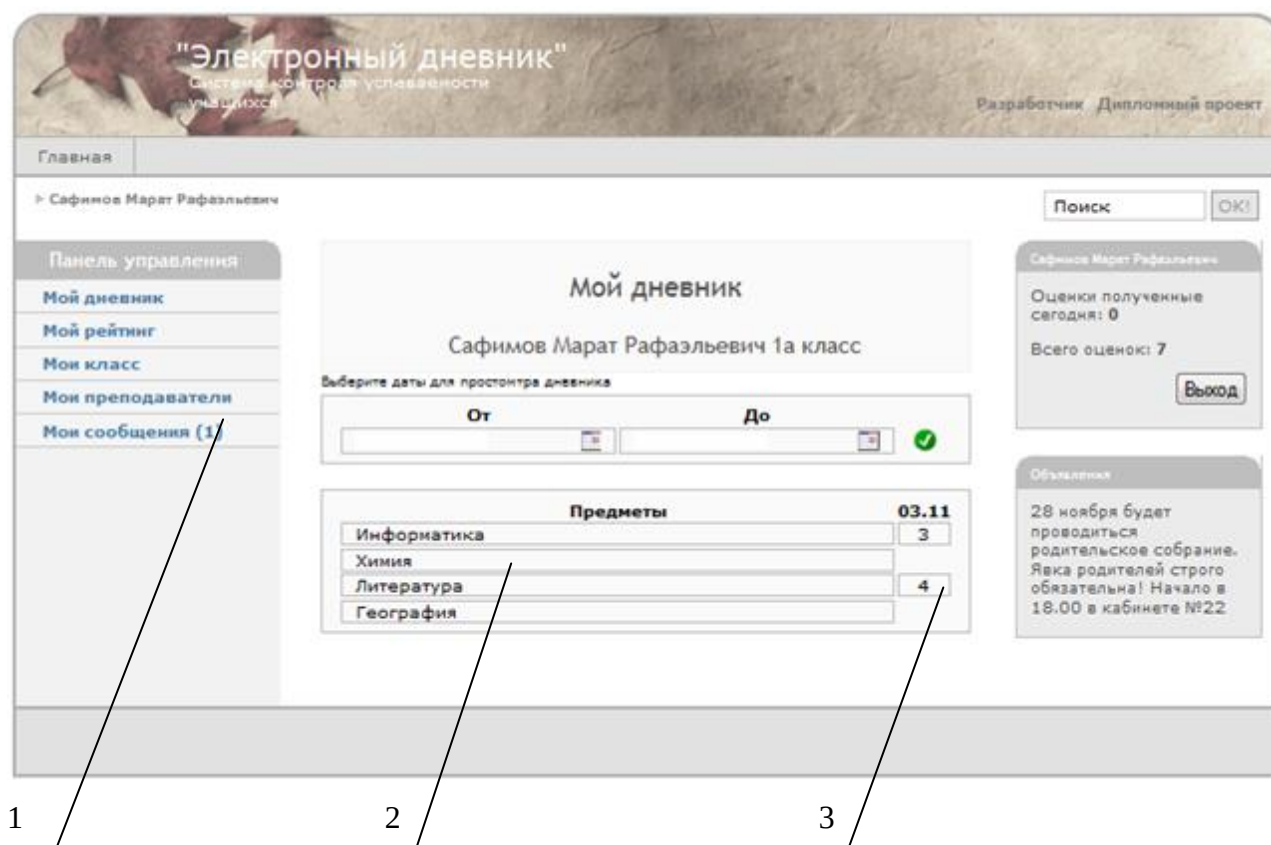


Рисунок 2.19– Окно просмотра оценок в дневнике

Таблица 2.28– Элементы окна просмотра оценок в дневнике

Обозначение элемента	Компонент	Описание
1	Href	Меню панели управления учащегося
2	Table	Список дисциплин группы
3	Table	Оценки

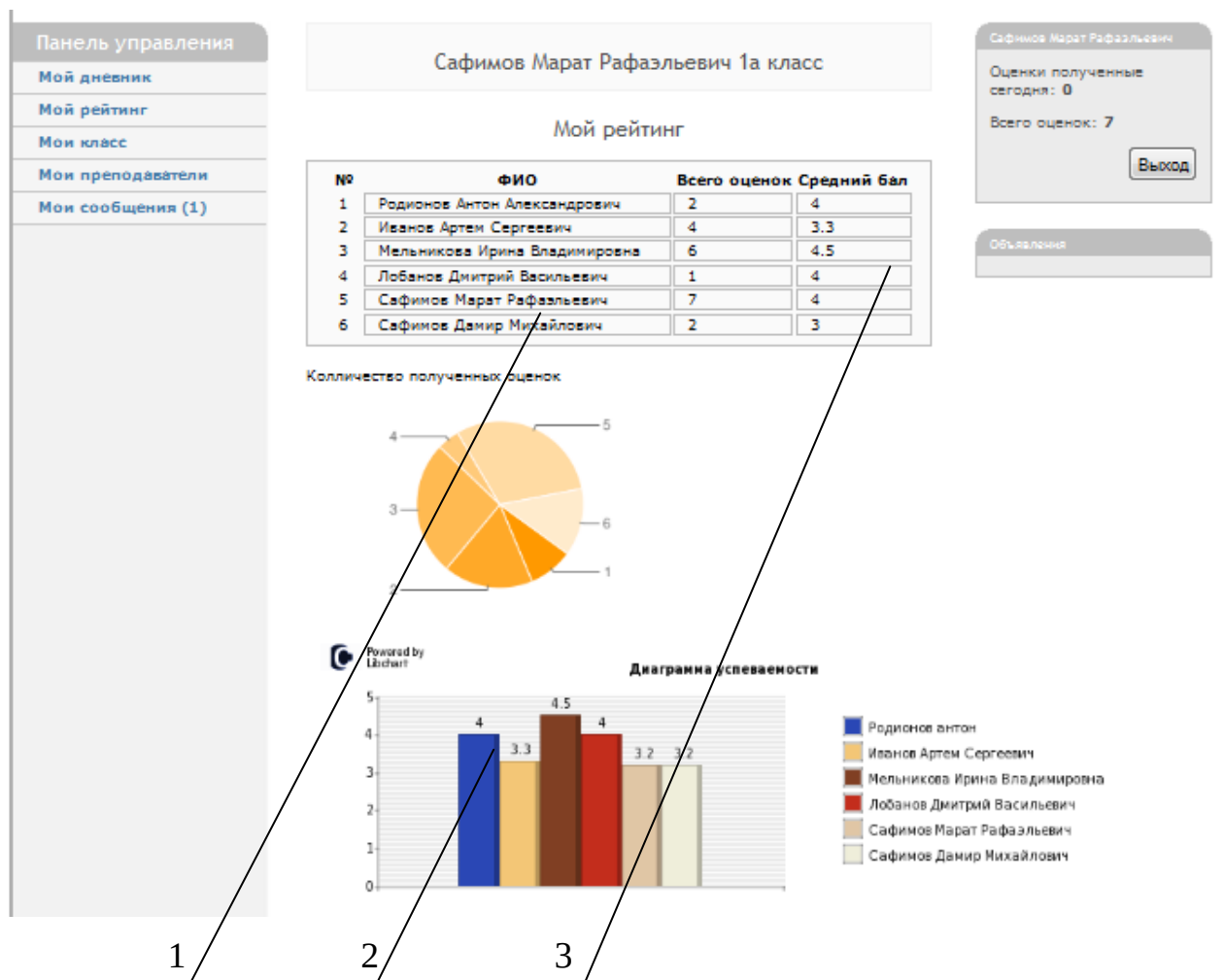


Рисунок 2.20– Окно просмотра рейтинга группы

Таблица 2.29– Элементы окна просмотра рейтинга группы

Обозначение элемента	Компонент	Описание
1	Table	Список группы
2	Chart	Диаграмма
3	Table	Средний бал

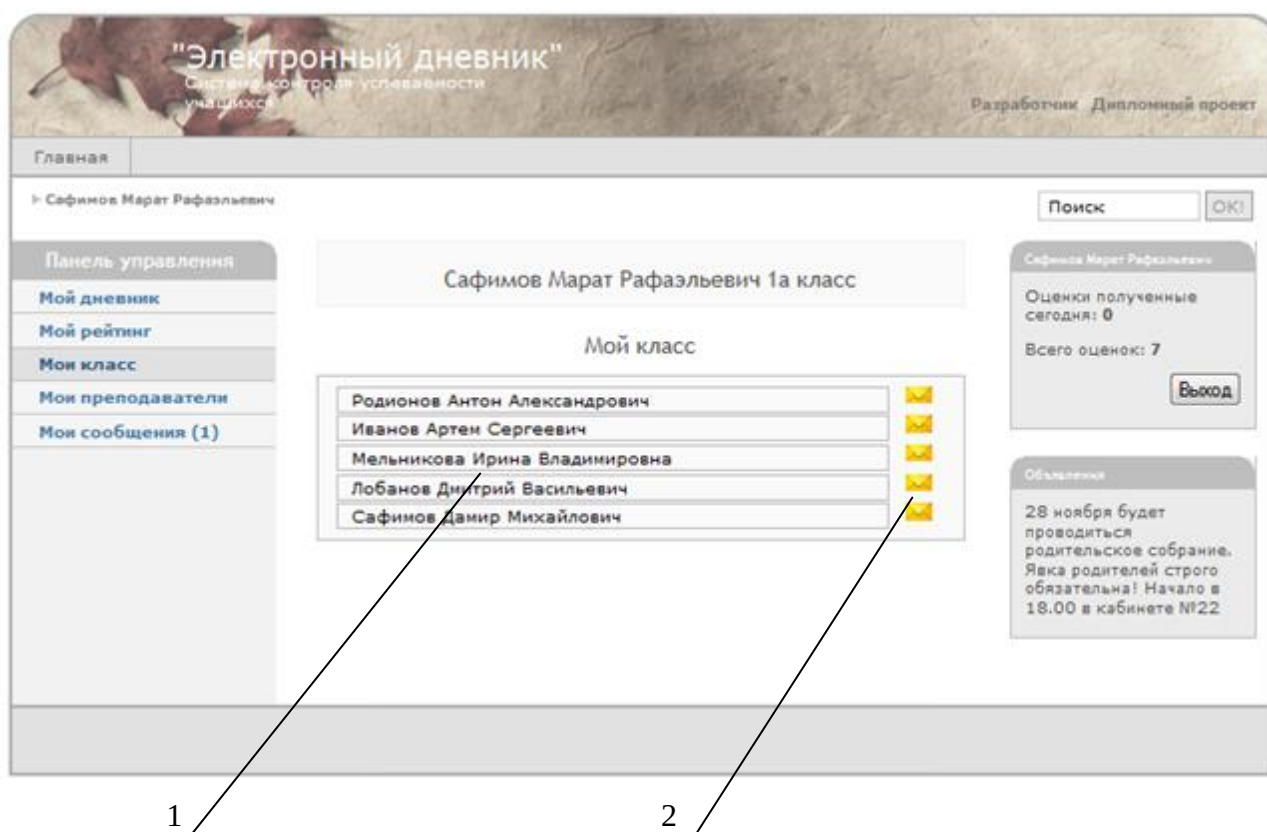


Рисунок 2.21 – Окно меню «Мой класс»

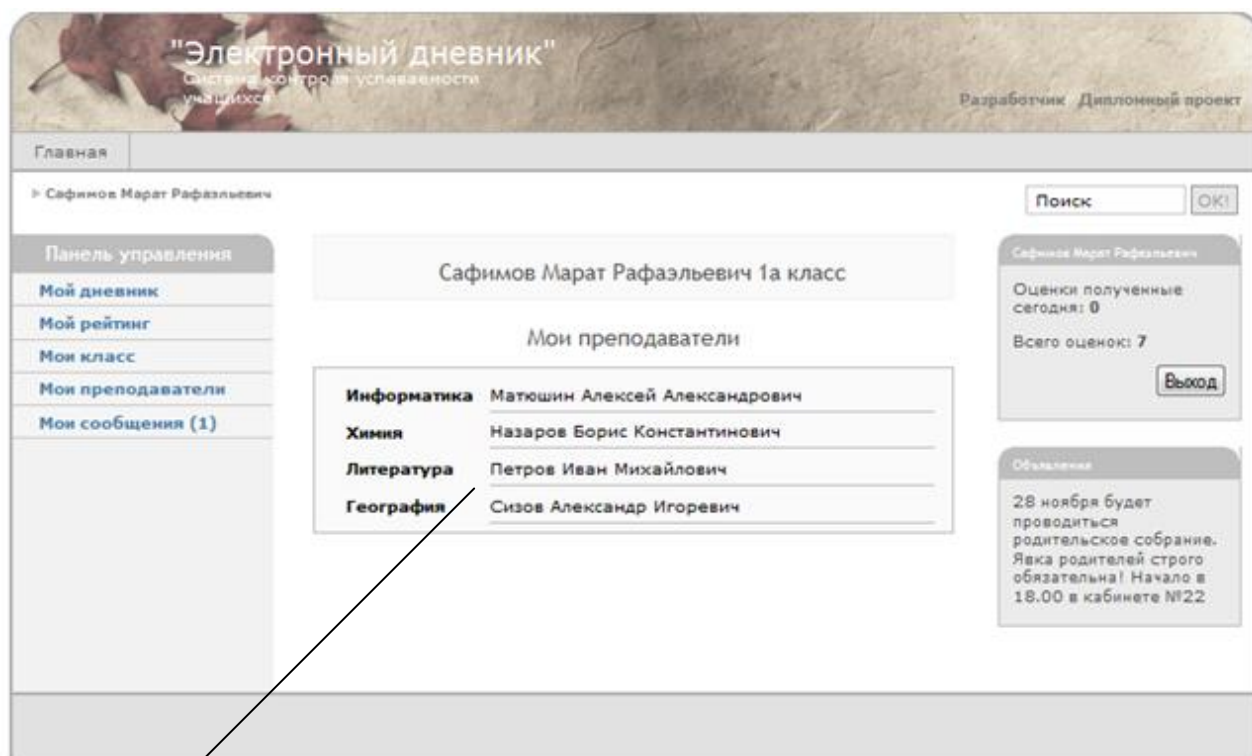
Таблица 2.30 – Элементы окна меню «Мой класс»

Обозначение элемента	Компонент	Описание
1	Table	Список группы
2	Button	Кнопка отправки сообщения однокласснику

Рисунок 2.22 – Окно добавления сообщения одноклассникам

Таблица 2.31– Элементы окна добавления сообщения одноклассникам

Обозначение элемента	Компонент	Описание
1	Form	Форма отправки сообщения
2	textarea	Поле ввода сообщения
3	Button	Кнопка отправки сообщения однокласснику



1

Рисунок 2.23 – Окно меню «Мои преподаватели»

Таблица 2.32– Элементы окна меню «Мои преподаватели»

Обозначение элемента	Компонент	Описание
1	Table	Список преподавателей и дисциплин

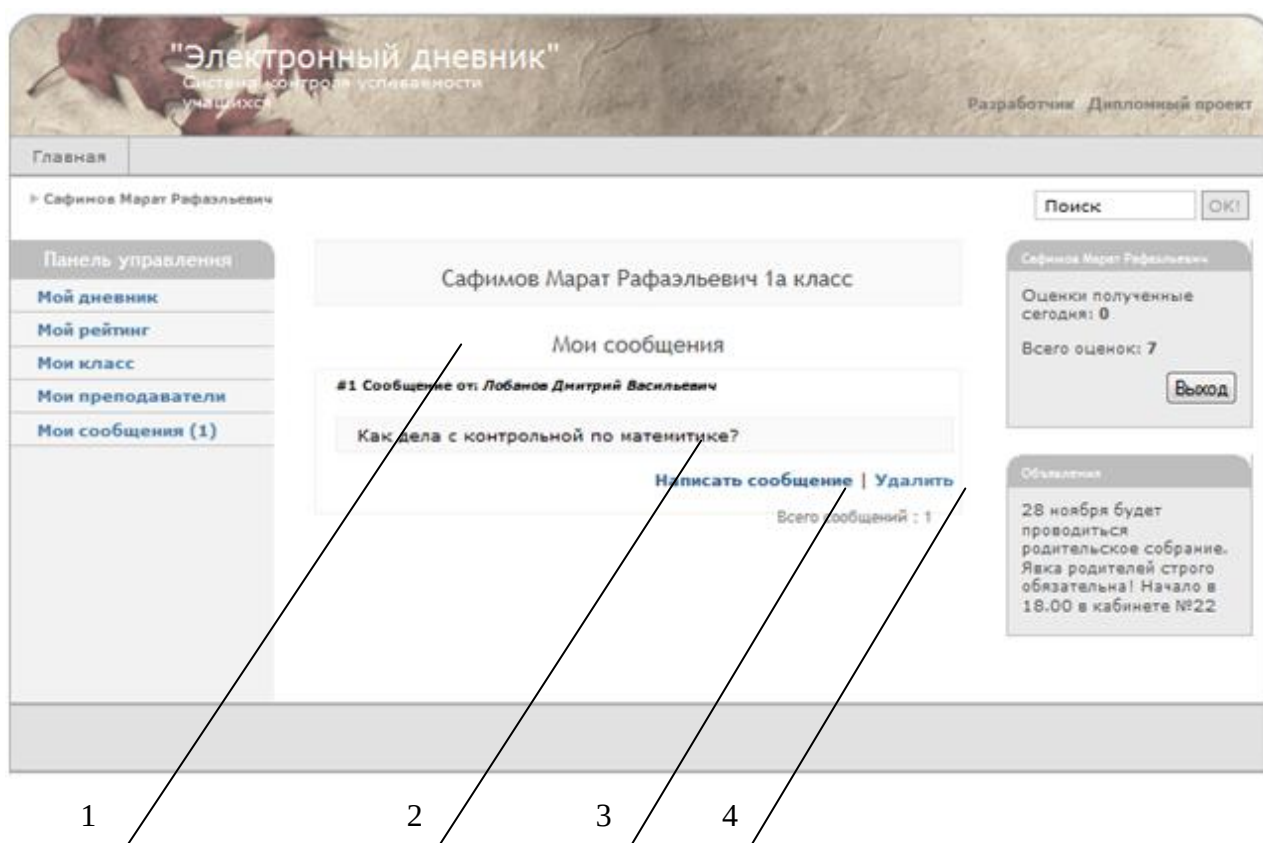


Рисунок 2.24 – Окно меню «Мои сообщения»

Таблица 2.33– Элементы окна меню «Мои сообщения»

Обозначение элемента	Компонент	Описание
1	Table	Информационное поле
2	Table	Принятые сообщения
3	Href	Ссылка на добавление сообщения
4	Href	Ссылка на удаление сообщения

2.5.1 Реализация

Программное средство разрабатывалось на языке программирования PHP.

Исходный код представлен в приложении А.

Рассмотрим спецификацию отдельных функций.

Имя: `mysql_connect()`

Назначение: устанавливает соединение с сервером MySQL. Следующие значения по умолчанию установлены для отсутствующих параметров: `server = 'localhost:3306'`, `username` = имя пользователя и `password` = пароль.

Примерреализации:

```
$server="localhost";  
$user="root";  
$passs="";  
$DB="dnevnik";  
mysql_connect($server,$user,$passs) or  
die("НетподключениякБД !!!");  
mysql_select_db($DB);
```

Имя: `mysql_select_db()`

Назначение: выбирает для работы указанную базу данных на сервере, на который ссылается переданный указатель. Если параметр указателя опущен, используется последнее открытое соединение. Если нет ни одного открытого соединения, функция попытается соединиться с сервером аналогично функции `mysql_connect()`, вызванной без параметров.

Имя: `mysql_query()`

Назначение: посылает запрос активной базе данных сервера. Если открытые соединения отсутствуют, функция пытается соединиться с СУБД, аналогично функции `mysql_connect()` без параметров. Результатзапросабуферизируется.

Примерреализации:

```
$count_class=mysql_query("SELECT * FROM groups,prepods  
WHERE prepods.school_id='".$$_SESSION['school_id']."'   
and groups.prepod_id=prepods.prepod_id");
```

Имя: mysql_num_rows()

Назначение: возвращает количество рядов результата запроса.

Примерреализации:

```
$mc=mysql_query("SELECT * FROM message WHERE  
in_id='".$$_SESSION[user_id]."'");  
if (!$mc) echo mysql_error();  
$mcl=mysql_num_rows($mc);
```

Имя: mysql_error()

Назначение: возвращает строку, содержащую текст ошибки выполнения последней функции MySQL, или (пустая строка) если операция выполнена успешно. Если в функцию не передан параметр ссылки на соединение, последнее открытое соединение будет использовано.

Пример реализации:

```
if (!$mc) echomysql_error();
```

Имя: mysql_fetch_array()

Назначение: возвращает массив с обработанным рядом результата запроса, или FALSE, если рядов больше нет.

Примерреализации:

```
$school=mysql_fetch_array(mysql_query("SELECT * FROM  
schools WHERE  
school_id='".$$_SESSION['school_id']."'"));
```

Имя: mysql_close()

Назначение: закрывает соединение с базой данных MySQL. Возвращает TRUE в случае успешного завершения или FALSE в случае возникновения ошибки.

3 Технология разработки и отладки web-приложений в среде NetBeans

Для реализации данного дипломного проекта была выбрана среда разработки NetBeans.

Эта среда — бесплатная, быстрая, кросс-платформенная, хорошо понимает язык PHP. В ней приятно писать PHP-код, потому что присутствует все необходимые для этого инструменты — подсветка, completion из всех подключенных файлов, браузер классов PHP, всплывающая документация по стандартным функциям и так далее. Вместе с тем, это не просто текстовый редактор. NetBeans умеет скачивать и закачивать файлы на удаленные сервера по FTP и SFTP, имеет развитый SQL-терминал и полностью автоматизирует работу с SVN или CVS.

3.1 Создание проекта в NetBeans

Ниже описана технология создания проекта в среде Eclipse.

Первый шаг, открытие помощника создания проекта - File > New > Project > PHP > PHP Application (рисунок 3.1).

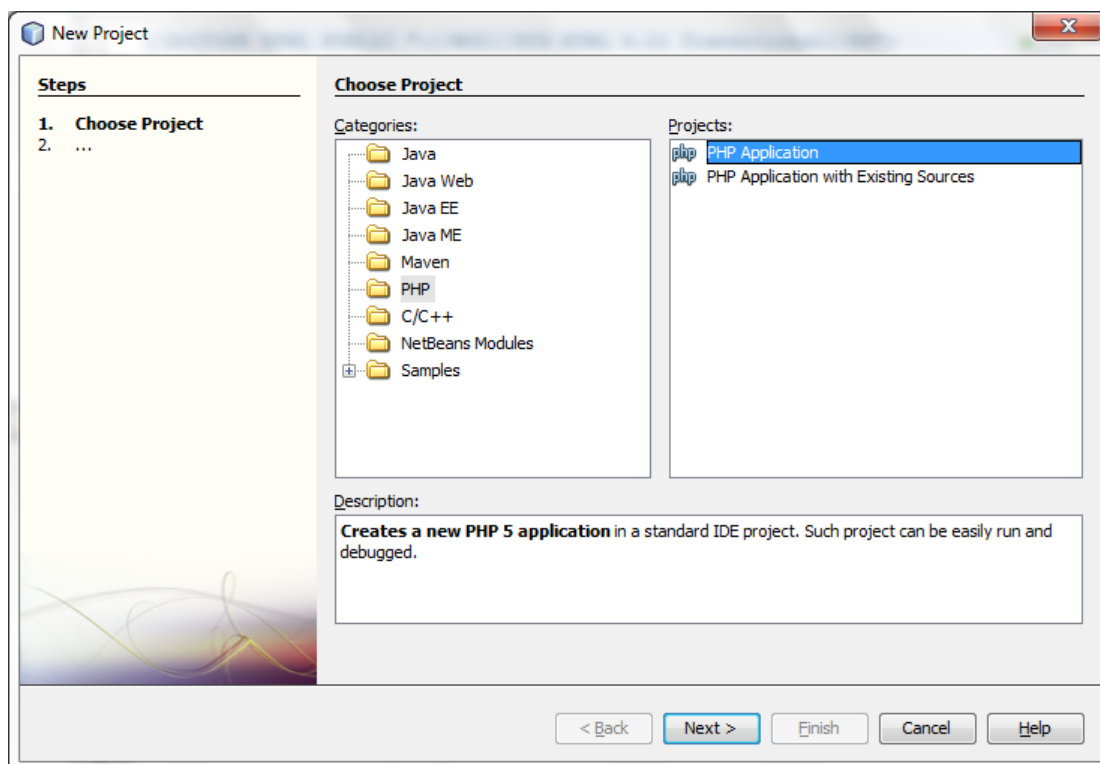


Рисунок 3.1 – Создание нового проекта

Нажмите кнопку Next и перейдите в диалог настройки пространства имен проекта (рисунок 3.2).

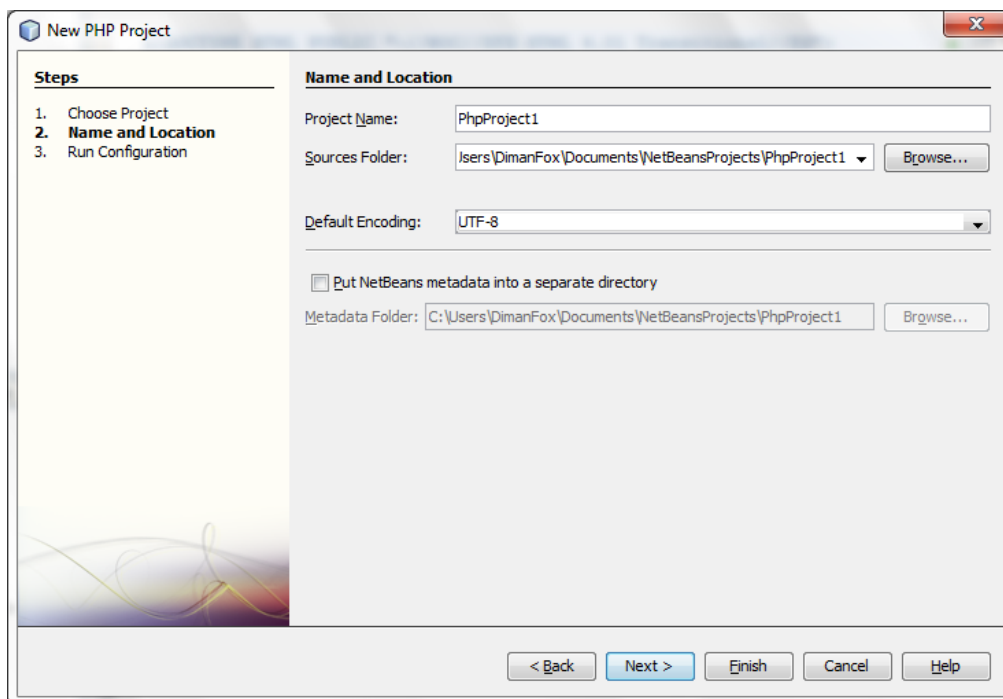


Рисунок 3.2 – Настройка пространства имен проекта

Нажмите кнопку Next и перейдите в диалог конфигурации проекта (рисунок 3.3).

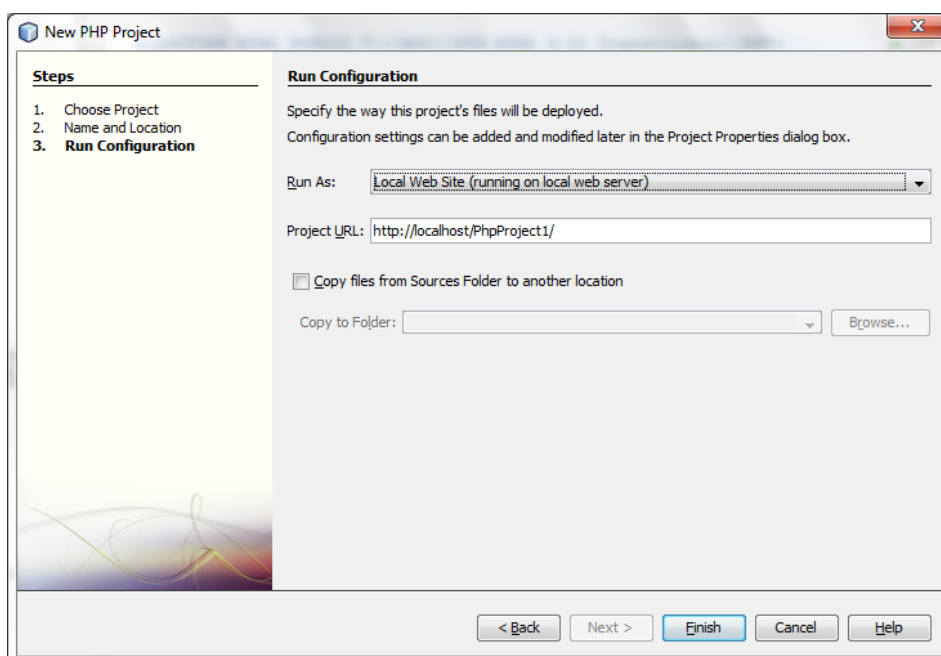


Рисунок 3.3 – Настройка конфигурации проекта

В результате вышеперечисленных операций появляется окно нового проекта, в котором присутствует структура каталога проекта и сгенерированные автоматически файлы настройки (рисунок 3.4).

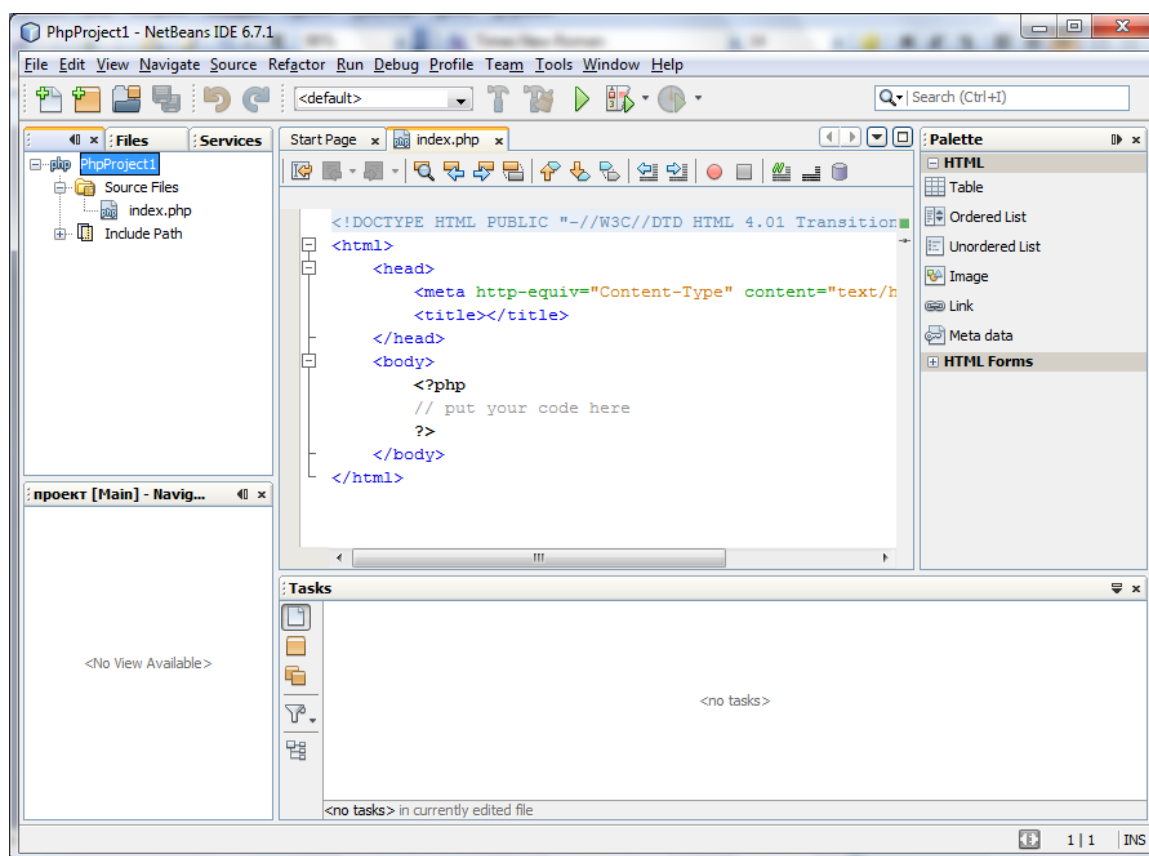


Рисунок 3.4 – Окно нового проекта

По умолчанию в проект добавлен файл `index.php`, который является главным файлом проекта, и к которому будут подключаться остальные файлы.

Для добавления новых файлов в проект нужно воспользоваться пунктом меню - File > New > New File (рисунок 3.5).

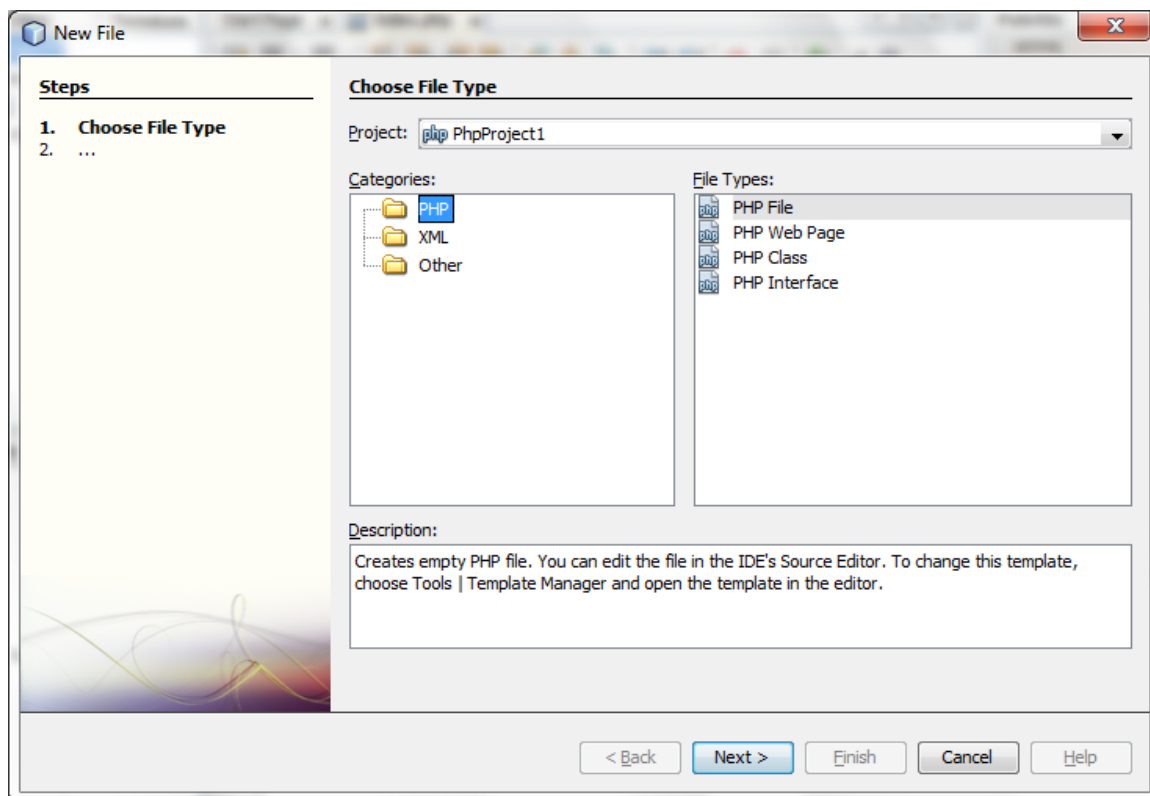


Рисунок 3.5 – Добавление нового файла в проект

После того как выбрали тип файла, который нужно добавить в проект, следует нажать кнопку «Next» для того чтобы задать имя файлу (рисунок 3.6).

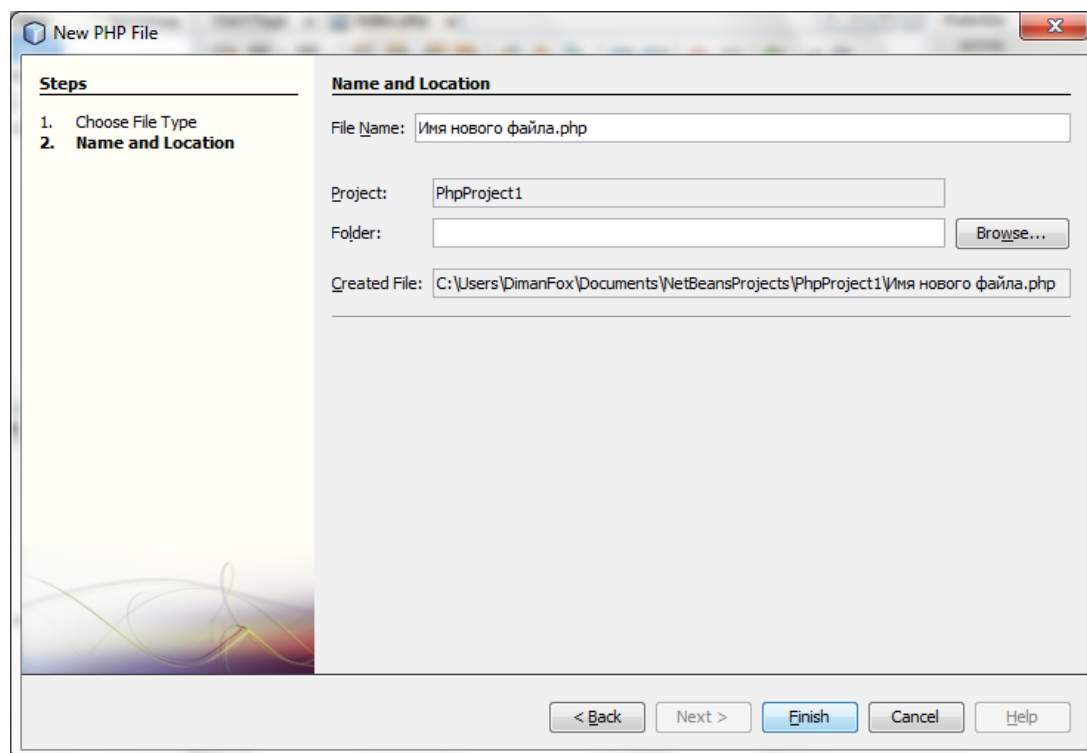


Рисунок 3.6 – Добавление нового файла в проект

После того как задали имя новому файлу, можно нажать кнопку «Finish» и новый файл добавиться в проект с минимум кода-шаблоном (рисунок 3.7).

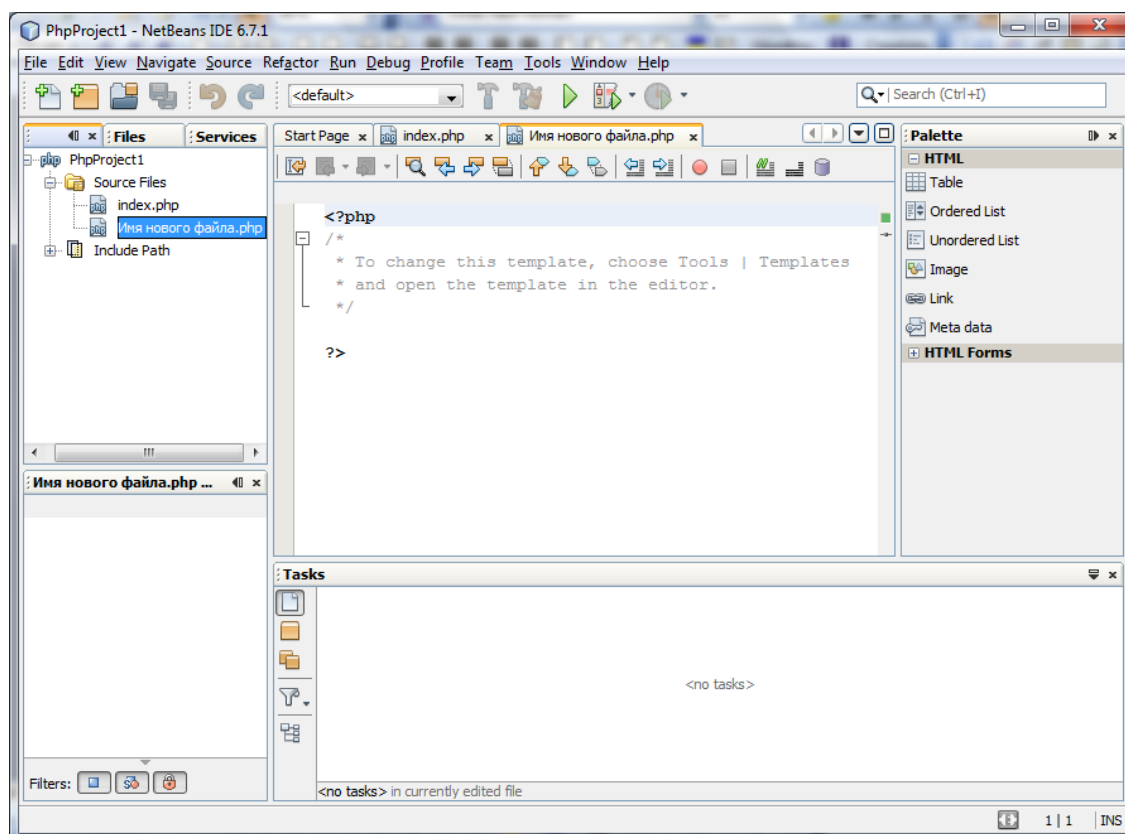


Рисунок 3.7 – Окно проекта

3.2 Отладка кода с помощью XDebug

Большинство разработчиков используют для отладки связку `echo` (`print_r`, `var_dump`) и `exit` (`die`), переходя от одной строке к другой. Однако, если ошибка появляется вновь в этом файле, требуется заново прописывать отладочные команды. Можно конечно отладку обернуть внутри конструкции `if`, которая будет работать только, когда определена какая-либо константа, например, `DEBUG`.

Xdebug позволяет создавать логи трассировки, достаточно хороший выход для данной ситуации. Однако лог трассировки, особенно когда он создан для части программы, предоставляет нам много информации, не имеющей никакого отношения к отладке. Таким образом, использование отладчика - это оптимальное решение. Отладчик позволит вам остановить запуск программы на некоторое время, проверить или модифицировать текущее значение

переменной, а затем продолжить выполнение программы. Запуская программу шаг за шагом, вы можете близко взглянуть как выполняется ваш код, что поможет найти вам места ошибок.

В среде разработки NetBeans IDE для отладки PHP кода есть очень удобное решение – Xdebug. Для его настройки необходимо:

- в соответствии с версией php, выбрать и скачать библиотеку xdebug.
- скопировать файл `php_xdebug.dll` в папку `/php/ext/`.

В файле `php.ini` добавить секцию XDebug.

```
[XDebug]
extension=php_xdebug-2.0.5-5.2.dll
zend_extension_ts="C:\server\php\ext\php_xdebug-2.0.5-5.2.dll"
xdebug.remote_enable=on
xdebug.remote_handler=dbgp
xdebug.remote_host=localhost
xdebug.remote_port=9000
```

После этого необходимо перезапустить Apache.

В NetBeans IDE перейти в Tools-Options-PHP. На вкладке General в секции Command Line указать путь до `php.exe`. В секции Debugging прописать port 9000 и Session ID `netbeans-xdebug` (рисунок 3.8).

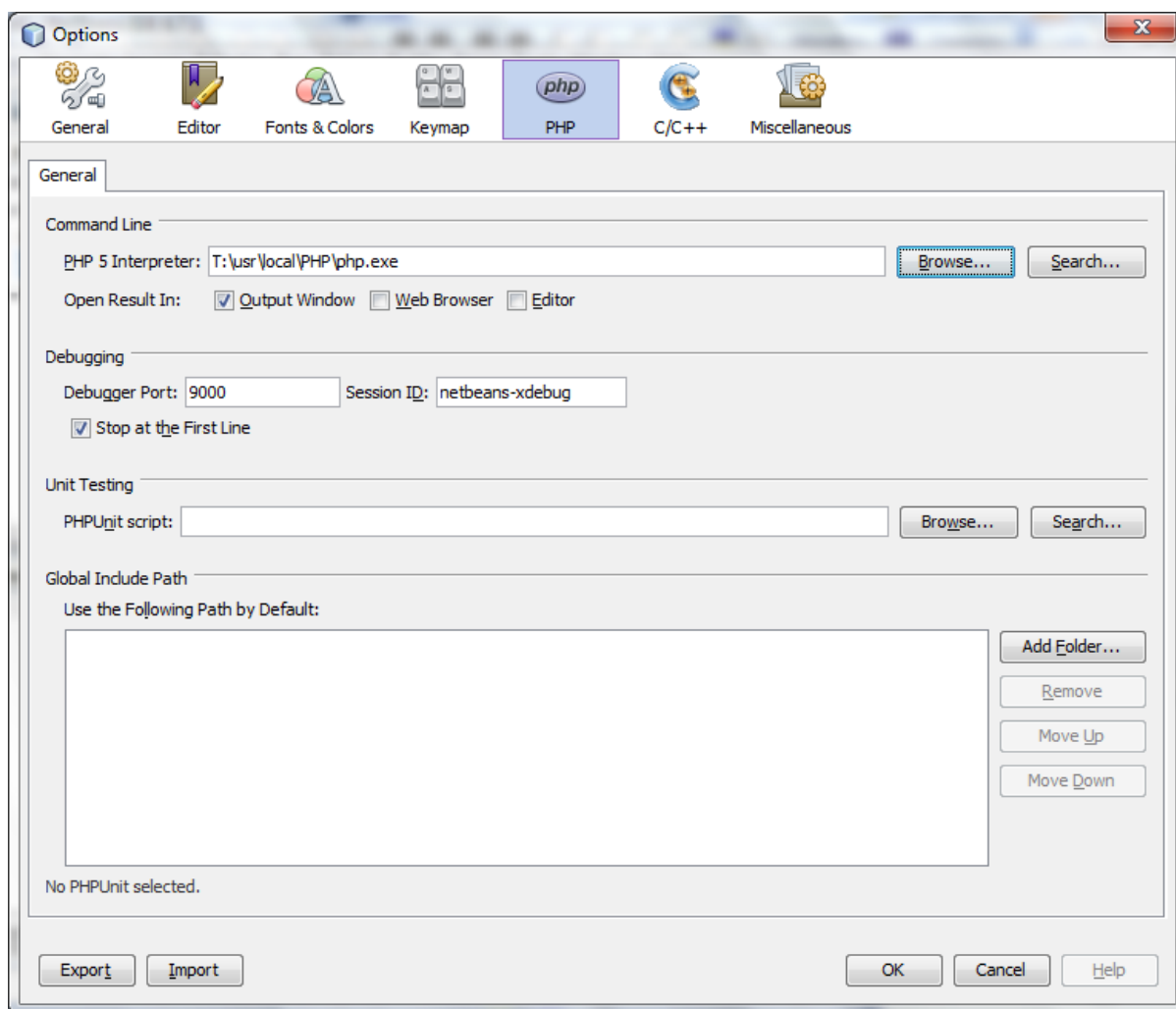


Рисунок 3.8 – Настройка отладчика

Когда отладка включена в `php.ini`, `xdebug` начинает контролировать выполнение программы, это значит, что `xdebug` может останавливать выполнение программы в любой момент, а потом запускать ее снова.

Когда выполнение программы остановлено, `xdebug` может получить информацию о текущем состоянии программы, например, прочитать текущие значения переменных. Также `xdebug` представляет возможность изменять значения переменных во время выполнения скрипта.

Расширение `xdebug` – это сервер, ожидающие соединений с клиентами на определенном порту, задаваемом в конфигурации. Существует два протокола, которые могут быть использованы для коммуникации между `xdebug`-клиентом и `xdebug`-сервером (GDB и DBGp).

GDB – это старый протокол, который был заменен на DBGp. Посылая команды серверу-xdebug, xdebug-клиент удаленно контролирует выполнение, сообщает PHP о приостановке запуска, выполнении следующей строки кода или продолжении выполнения программы. Клиент обычно встроен в редактор или IDE (в нашем случае в NetBeans), поэтому вы не будете иметь дело с протоколом общения напрямую.

Веб-сервер с xdebug может быть запущен на другой операционной системе, нежели xdebug-клиент. Вот почему xdebug называется удаленным отладчиком (remote debugger). Для упрощения, мы настроим сервер и клиент на одном и том же компьютере.

Существует два режима начала сессии отладки. Эти режимы контролируются в `php.ini` настройкой `xdebug.remote_mode`. Значение по умолчанию `req`, xdebug-клиент будет соединяться с сервером всякий раз, когда начинается выполнение скрипта. Если вы хотите, чтобы xdebug соединялся с сервером только тогда, когда установлен breakpoint или, когда в скрипте возникла ошибка, то можно установить `xdebug.remote_mode` в `jit`.

Для того чтобы начать отладку, нужно послать параметр `XDEBUG_SESSION_START` в скрипт как часть GET, POST или COOKIE. Значение этого параметра - это название сессии отладки, которое должно быть уникальным, по имени сессии xdebug может различать различные сессии, запущенные параллельно. Для завершения сессии отладки необходимо послать скрипту команду `XDEBUG_SESSION_STOP`.

Вместо того чтобы вручную начинать и заканчивать отладочные сессии, следует установить специальный `firefox plugin`, который поможет легко начинать и заканчивать сессии одним нажатием мышки.

ЗАКЛЮЧЕНИЕ

В результате выполнения задания на магистерскую диссертацию было разработано программное обеспечение для создания системы контроля успеваемости учащихся.

Разработка проведена в операционной системе Windows 10 на персональном компьютере совместимом с IBM PC.

В данной работе были рассмотрены основные принципы организации учебного процесса и контроля качества знаний учащихся с использованием базальной системы, спроектирована база данных, которая позволяет удобно пользоваться необходимой информацией. Также был написан Веб-интерфейс на языке PHP. Для хранения информации использовалась база данных MySQL.

В ходе выполнения работы были выполнены следующие задачи:

- изучение особенностей организации и управления данными в информационных системах,
- вывод и манипулирование данными для формирования диаграмм и статистической информации,
- обработка статистической информации,
- проектирование Web-интерфейса,
- реализация физической модели базы данных с помощью языка запросов SQL,
- создание возможности администрирования системы,
- создание динамического web-сайта.

Осуществлено функциональное тестирование разработанного приложения, которое показало корректность его работы.

При выполнении магистерской диссертации использовался язык моделирования UML.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Буранов В. А. Основы программирования на PHP. - М.: Изд-во «Информационные технологии», 2021. — 268 с.
2. Брин Ф. М. LAMP. Основы и технологии - М: «Символ-Плюс», 2020 г. - 504 с.
3. Ватсон К. Скрипты и шаблоны JavaScript. - М.: Лори, 2019. - 480 с.
4. Вирт Н. Алгоритмы и структуры данных. - Спб: Невский диалект, 2017 г. - 352 с.
5. Гирасименко П. В. Платформа WEB для программистов. - СПб.: Питер, 2018. - 656 с.
6. Габышев А. И. Архитектура современных WEB серверов. - М: Бином, 2018 г. - 272с.

ПРИЛОЖЕНИЕ А

ЛИСТИНГ ИСХОДНОГО КОДА

Index.php

```
<?
session_start();
error_reporting(E_USER_NOTICE);
//error_reporting(E_ALL);
include "db/connect.php";
if($_GET['page']!="")
{
    if($_GET['sect']!="") {
        if(file_exists("pages/".$_GET['sect']."/".$_GET['page'].".php")) {
            $include="pages/".$_GET['sect']."/".$_GET['page'].".php";
        }else{
            $include="pages/404.php";
        }
    }else{
        if(file_exists("pages/".$_GET['page'].".php")) {
            $include="pages/".$_GET['page'].".php";
        }else{
            $include="pages/404.php";
        }
    }
}else{
    $include="pages/index.php";
}

?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en">
<head>

<link rel="stylesheet" type="text/css" media="screen,projection,print"
href="./css/mf42_layout4_setup.css" />
<link rel="stylesheet" type="text/css" media="screen,projection,print"
href="./css/mf42_layout4_text.css" />
<link rel="icon" type="image/x-icon" href="./img/favicon.ico" />
<title>Дипломный проект: Система контроля успеваемости учащихся</title>
</head>

<!-- Global IE fix to avoid layout crash when single word size wider than column
width -->
<!--[if IE]><style type="text/css"> body {word-wrap: break-
word;}</style><![endif]>-->

<body>
<!-- Main Page Container -->
<div class="page-container">

<!-- START COPY here -->

<!-- A. HEADER -->
<div class="header">

<!-- A.1 HEADER TOP -->
<div class="header-top">
<div class="round-border-topleft"></div><div class="round-border-
topright"></div>

<!-- Sitelogo and sitename -->
<a class="sitelogo2" href="index.php" title="Home"></a>
<div class="sitename">
<h1><a href="index.php" title="Перейти на главную">"Электронный дневник"</a></h1>
```

```

<h2>Система контроля успеваемости учащихся</h2>
</div>

<!-- Navigation Level 1 -->
<div class="nav1">
<ul>
<li><b><a href="index.php?page=about"
title="Оразработчике">Разработчик</a></b></li>
<li><b><a href="#" title="О дипломном проекте">Дипломный проект</a></b></li>

        </ul>

</div>
</div>

<!-- A.3 HEADER BOTTOM -->
<div class="header-bottom">

<!-- Navigation Level 2 (Drop-down menus) -->
<div class="nav2">

<!-- Navigation item -->
<ul>
<li><a href="index.php">Главная</a></li>
</ul>

<!-- Navigation item -->

</li>
</ul>
</div>
    </div>

<!-- A.4 HEADER BREADCRUMBS -->

<!-- Breadcrumbs -->

<div class="header-breadcrumbs">

    <?
        if (isset($_SESSION['access'])) {
            echo ("<ul>
<li>");
                echo $_SESSION['user_name'];
            echo ("</li>
</ul>");
        }

    ?>

<!-- Search form -->
<div class="searchform">
<form action="index.html" method="get">
<fieldset>
<input name="field" class="field" value=" Поиск" />
<input type="submit" name="button" class="button" value="OK!" />
</fieldset>
</form>
</div>
</div>
</div>

```

```

<!-- END COPY here -->

<!-- B. MAIN -->
<div class="main">

<!-- B.1 MAIN NAVIGATION -->
<?
    if (isset($_POST['school_login'])) {

if (isset($_POST['school_psw']))
    {
        $err='';
        $pass="SELECT * FROM schools WHERE
school_psw='".$_POST['school_psw']."'";
        $res=mysql_query($pass);
        if ((mysql_num_rows($res))>0) {
            $result = mysql_fetch_array($res);
            $_SESSION['access']='3';
            $_SESSION['school_name']=$result['school_name'];
            $_SESSION['school_id']=$result['school_id'];
            $_SESSION['user_name']=$result['school_dir_name'];
            echo "<meta http-equiv='refresh' content='0';
URL=index.php?sect=schools&page=index'>";
            } else $err3="Ошибочный код доступа";
        }
    }
    ?>
    <div class="main-navigation">

<!-- Navigation Level 3 -->
<div class="round-border-topright"></div>

    <? if ($_SESSION['access']==3) include "pages\menu_school.php";?>
    <? if ($_SESSION['access']==2) include "pages\menu_prepod.php";?>
    <? if ($_SESSION['access']==1) include "pages\menu_student.php";?>
    <? if (!isset($_SESSION['access'])) { ?>
    <h1 class="first" align='center'>Общая информация:</h1>
    <?
    $count_schools_res=mysql_query("SELECT * FROM schools");
    $count_schools_result= mysql_num_rows($count_schools_res);
    $count_prepod_res=mysql_query("SELECT * FROM prepods");
    $count_prepod_result= mysql_num_rows($count_prepod_res);
    $count_rite_res=mysql_query("SELECT * FROM journal");
    $count_rite_result= mysql_num_rows($count_rite_res);
    ?>
    <p><b>Зарегистрировано:</b></p>
    <p>Школ: <b><?echo $count_schools_result?></b></p>
    <p>Преподавателей: <b><?echo $count_prepod_result?></b></p>
    <p>Выставлено оценок: <b><?echo $count_rite_result?></b></p>

    <h1 class="first" align='center'>Что требуется от школы для
подключения?</h1>
    <p>1) Необходимо назначить администратора, который будет подключать новых
пользователей и управлять работой школы в Дневнике.</p>
    <p>2) Не требуется ни установки программного обеспечения, ни закупки
дополнительного оборудования. Работать в Дневнике можно с любого компьютера,
подключенного к сети Интернет.</p>
    <div class="block">
    <p><a
href='index.php?page=registration'>ЗАРЕГИСТРИРОВАТЬ ШКОЛУ в системе</a></p>
    </div>

```

```

        <div class="loginform">
            Вход для администрации школ:
        <form method="post">
        <p><label class="top">Код доступа</label><br>
            <input name="school_psw" class="field" type="password"></p>
            <?if (!empty($err3)) echo "<p align='center'><font
color='red'>". $err3."</font></p>";?>
            <p align="right"><input name="school_login" class="button" value="Ok"
type="submit"></p>
        </form>
        </div>
        <?}?>

<!-- Template infos -->

</div>

<!-- B.2 MAIN CONTENT -->
<div class="main-content">

<?
        if($include!="")
        {
            include($include);
        }
        else
        {
include("pages/index.php");
        }
        ?>

</div>

<!-- B.3 SUBCONTENT -->

<?include "pages/right_panel.php";?>

        </div>

<!-- C. FOOTER AREA -->

<div class="footer">
</div>
</div>

</body>

</html>

```

Pages/index.php

```

<!-- Pagetitle -->
<h1 class="pagetitle" align='center'>Добро пожаловать в интерактивную
систему<br>"Электронный дневник"</h1>

<!-- Content unit - One column -->
<divclass="column1-unit">
    <h1>Система «Электронный дневник школьника на основе web-
технологий»</h1>
    <div class='img_left'></div>
    <p>предназначена, главным образом, для того, чтобы родители
современных школьников могли осуществлять контроль успеваемости своих детей в
любом месте, где есть доступ к всемирной сети Internet. А также для учителей, с

```

целью облегчения выставления оценок по своим предметам (например, проверяя контрольные работы, находясь дома).</p>

<p>Система помогает отобразить реальную картину успеваемости, как для учителей, так и для учащихся и их родителей. Информация о каждом учащимся является конфиденциальной. Для входа в систему необходимо указать свой личный пароль. Интерактивная система является удобным инструментом для создания единого информационно-образовательного пространства взаимодействия образовательного учреждения и родителей учащихся.</p>

<hr class="clear-contentunit" />

</div>

Pages/menu_school.php

<h1 class="first" align='center'>Панель управления</h1>

<!-- Navigation with grid style -->

<dl class="nav3-grid">

<dt>Статистика</dt>

<dt>Классы</dt>

<dt>Преподаватели</dt>

<dt>Рейтинги</dt>

<dt>Поиск</dt>

<dt>Настройки</dt>

</dl>

Pages/menu_student.php

<h1 class="first" align='center'>Панель управления</h1>

<!-- Navigation with grid style -->

<dl class="nav3-grid">

<dt>Мой дневник</dt>

<dt>Мой рейтинг</dt>

<dt>Мои класс</dt>

<dt>Мои

преподаватели</dt>

<?>

\$mc=mysql_query("SELECT * FROM message WHERE
in_id=".\$SESSION[user_id].");

if (!\$mc) echo mysql_error();

\$mc1=mysql_num_rows(\$mc);

?>

<dt>Мои сообщения (<?echo
\$mc1;?>)</dt>

</dl>

Pages/menu_prepod.php

<h1 class="first" align='center'>Панель управления</h1>

<!-- Navigation with grid style -->

<dl class="nav3-grid">

<dt>Классы</dt>

<dt>Объявления</dt>

</dl>

Pages/right_panel.php

<div class="main-subcontent">

```

<?
if (isset($_POST['out']))
{
    session_destroy();
    echo "<meta http-equiv='refresh' content='0 url=index.php'>";
}

if (isset($_POST['u_login'])) {

if (isset($_POST['u_psw']))
{
    $err='';
    $pass="SELECT * FROM users, groups WHERE
users.user_psw='".$_POST['u_psw']."' and users.group_id=groups.group_id";
    $res=mysql_query($pass);
    if ((mysql_num_rows($res))>0) {
        $result = mysql_fetch_array($res);
        $_SESSION['access']='1';
        $_SESSION['user_name']=$result['user_name'];
        $_SESSION['user_id']=$result['user_id'];
        $_SESSION['group_id']=$result['group_id'];
        $_SESSION['school_id']=$result['school_id'];
        echo "<meta http-equiv='refresh' content='0;
URL=index.php?sect=student&page=index'>";
        } else $err1="Ошибочный код доступа";
    }
}
if (isset($_POST['p_login'])) {
if (isset($_POST['p_psw']))
{
    $err='';
    $pass="SELECT * FROM prepos WHERE
prepod_psw='".$_POST['p_psw']."'";
    $res=mysql_query($pass);
    if ((mysql_num_rows($res))>0) {
        $result = mysql_fetch_array($res);
        $_SESSION['access']='2';
        $_SESSION['user_name']=$result['prepod_name'];
        $_SESSION['user_id']=$result['prepod_id'];
        $_SESSION['school_id']=$result['school_id'];
        echo "<meta http-equiv='refresh' content='0;
URL=index.php?sect=prepod&page=index'>";
        } else $err2="Ошибочный код доступа";
    }
}

if (!isset($_SESSION['access'])) {

?>
<!-- Subcontent unit -->
<div class="subcontent-unit-border">
<div class="round-border-topleft"></div>
    <div class="round-border-topright"></div>
<h1 align='center'>Вход в дневник</h1>
    <h3>Для просмотра дневника учащегося</h3>
<div class="loginform">
<form method="post">
<p><label class="top">Код доступа</label><br>
    <input name="u_psw" class="field" type="password"></p>

```

```

        <?if (!empty($err1)) echo "<p align='center'><font
color='red'>". $err1."</font></p>";?>
        <p align="center"><input name="u_login" class="button" value="Вход
учащегося" type="submit"></p>
        <p>Код доступа можно получить у администрации школы</p>
        </form>
</div>

</div>

        <div class="subcontent-unit-border">
<div class="round-border-topleft"></div>
        <div class="round-border-topright"></div>
<h1 align='center'>Вход в журнал</h1>
        <h3>Для выставления оценок преподавателем</h3>
<div class="loginform">
<form method="post">
<p><label class="top">Код доступа</label><br>
        <input name="p_psw" class="field" type="password"></p>
        <?if (!empty($err2)) echo "<p align='center'><font
color='red'>". $err2."</font></p>";?>
        <p align="center"><input name="p_login" class="button" value="Вход
преподавателя" type="submit"></p>
        <p>Код доступа можно получить у администрации школы</p>
        </form>
</div>

</div>

        <?> else {?>
<!-- Subcontent unit -->
<div class="subcontent-unit-border">
<div class="round-border-topleft"></div>
        <div class="round-border-topright"></div>
<h1 style='font-size:100%;'><?echo $_SESSION['user_name'];?></h1>

        <?
        if ($_SESSION['access']==1) {
        $dt=date('Y-m-d');
        $d = explode("-", $dt);
        $v_data="$d[0]$d[1]$d[2]";
        $col_today=mysql_query("SELECT count(*) FROM journal WHERE
data=$v_data and user_id=".$_SESSION['user_id']."");
        $t=mysql_fetch_row($col_today);
        $col_all=mysql_query("SELECT count(*) FROM journal WHERE
user_id=".$_SESSION['user_id']."");
        $t_all=mysql_fetch_row($col_all);
        echo ("
        <p>Оценки полученные сегодня: <b>$t[0]</b></p>
        <p>Всего оценок: <b>$t_all[0]</b></p>
        ");
        }
?>

        <form method="POST">
        <div class="right_block">
        <p align="center"><input name="out" type="submit" value="Выход"></p>
        </div>
        </form>
</div>

        <?>?>
<!-- Subcontent unit -->
<?
        if ($_SESSION['access']==1) {

```

```

        echo("
        <div class='subcontent-unit-border'>
<div class='round-border-topleft'></div>
        <div class='round-border-topright'></div>
<h1 style='font-size:100%;'>Объявления</h1>
        ");
        $adv=mysql_query("SELECT * FROM advert WHERE
in_id=".$SESSION['group_id']." ORDER BY advert_id DESC");
        if (!$adv) echo mysql_error();
        while($my_adv=mysql_fetch_array($adv)){
            echo("<p>".$my_adv['advert']."</p>");
        }
        echo("
        </div>");
    }
    ?>
</div>

```

Pages\registration.php

```

<?
        function rndstr($length)
        {
            $return = $chars = null;
            $chars .= 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789';
            for($i = 0, $sl = strlen($chars) - 1; $i <= $length; $i++)
            $return .= $chars[rand(0, $sl)];
            return $return;
        }

        function test_reg_form($name,$adress,$tel,$dir,$code) {
            $_SESSION['err']="";
            if (empty($name))
            {
                $_SESSION['err']="Не заполнено поле \"Название\"";
                return (false);
            }
            if (empty($adress))
            {
                $_SESSION['err']="Не заполнено поле \"Адрес\"";
                return (false);
            }
            if (empty($tel))
            {
                $_SESSION['err']="Не заполнено поле \"Телефон\"";
                return (false);
            }
            if (empty($dir))
            {
                $_SESSION['err']="Не заполнено поле \"ФИО Директора\"";
                return (false);
            }
            if ($code!= $_SESSION['rand_code'])
            {
                $_SESSION['err']="КОД ПОДТВЕРЖДЕНИЯ НЕ СОВПАДАЕТ!!!";
                return (false);
            }
            $q=mysql_query("SELECT COUNT(*) FROM schools WHERE
school_name='".$$_POST['name_school']."' and
school_adress='".$$_POST['adress_school']."'");
            $r1=mysql_result($q,0,0);
            if ($r1!=0)
            {

```

```

$_SESSION['err']="Школа с таким именем уже зарегистрирована";
return (false);
}
return (true);
}

if (isset($_POST['reg_submit'])) {

    if
(test_reg_form($_POST['name_school'],$_POST['adress_school'],$_POST['tel_school']
,$_POST['dir_school'],$_POST['reg_code'])) {

        $t=true;
        while ($t):
            $code_access=rndstr(7);
            $res=mysql_query("SELECT * FROM schools WHERE
school_psw='".$_.$code_access."'");
            $rows=mysql_num_rows($res);
            if ($rows==0) $t=false;
        endwhile;

        $q=mysql_query("INSERT INTO schools
VALUES('','".$_.$POST['name_school']. "','".$_.$POST['adress_school']. "','".$_.$POST['t
el_school']. "','".$_.$POST['dir_school']. "','".$_.$code_access.')");
        if (!$q) echo ("Ошибка при регистрации... Попробуйте еще раз
позже!!!");
        else $good = "Школа успешно зарегистрирована!!! Ваш код доступа
\"".$_.$code_access.\"\"";
        }
        }
        ?>

        <h1 class="pagetitle" align='center'>Добро пожаловать в
интерактивную систему<br>"Электронный дневник"</h1>

<!-- Content unit - One column -->
<div class="column1-unit">
    <h1 align='center'>Регистрация школы в системе</h1>
    <hr class="clear-contentunit" />
<div class="contactform">
    <?
        if (!empty($_SESSION['err'])) echo ("<center><font
color='red'> ".$_.$SESSION['err']. "</font></center>");
        if (!empty($good)) { echo ("
<div class='infoblock'>
            ".$good."
        </div>
        ");
        }

        else {
            ?>

<form method="post">
<fieldset><legend>&nbsp;&nbsp;&nbsp;Информация о школе&nbsp;&nbsp;&nbsp;</legend>

            <p>
                <label class="left">Название</label>
<input type="text" name="name_school" class="field"/>
                <h3>Пример: Средняя образовательная школа №39</h3>
            </p>

```

```

        <p><label class="left">Адрес:</label>
<input type="text" name="adress_school" class="field"/>
        <h3>Пример: г.Пенза ул. Пр.Победы 21</h3>
        </p>
        <p><label class="left">Телефон:</label>
<input type="text" name="tel_school" class="field"/>
        <h3>Пример: 8 (412) 571582</h3>
        </p>
        <p><label class="left">ФИО директора</label>
<input type="text" name="dir_school" class="field"/>
        <h3>Пример: Иванов Иван Иванович</h3>
        </p>
        <p><label class="left">Код:</label>
<input type="text" name="reg_code" class="field"/>
        </p>

<p><input type="submit" name="reg_submit" class="button"
value="РЕГИСТРАЦИЯ"/></p>
</fieldset>
</form>

        <?}?>

</div>

        <?

        ?>

</div>

```

Pages/captcha.php

```

<?php

session_start();

$string = '';

for ($i = 0; $i < 5; $i++) {
    // this numbers refer to numbers of the ascii table (lower case)
    $string .= chr(rand(97, 122));
}

$_SESSION['rand_code'] = $string;

$dir = '../fonts/';

$image = imagecreatetruecolor(170, 60);
$black = imagecolorallocate($image, 0, 0, 0);
$color = imagecolorallocate($image, 200, 100, 90); // red
$white = imagecolorallocate($image, 255, 255, 255);

imagefilledrectangle($image, 0, 0, 399, 99, $white);
imagefttext ($image, 30, 0, 10, 40, $color, $dir."arial.ttf",
$_SESSION['rand_code']);

header("Content-type: image/png");
imagepng($image);

?>

```

Pages/prepod/journal.php

```
<script language="JavaScript" src="js/calendar_us.js"></script>
<link rel="stylesheet" href="css/calendar.css">

<?

if (isset($_POST['go']))
{
    $err_="";
    for ($i=0; $i<sizeof($_POST['fname']); $i++)
    {
        if ((($_POST['fname'][$i]>1) && ($_POST['fname'][$i]<6))
        {
            $query_res=mysql_query("INSERT INTO journal
VALUES('','"$_POST['dates']."'','"$_GET['g_id']."'','"$_POST['user'][$i]."',".$_P
OST['fname'][$i].")");
        } else $err_="Не все оценки были добавлены!";
        //echo "ocenka".$_POST['fname'][$i];
        //echo "User".$_POST['user'][$i];
    }
    if ($err_!="") {
        echo $err_;
    } elseif ("База данных была обновлена!");
}

?>
<div class="column1-unit">

<?$query_group=mysql_query("SELECT * FROM predmet_groups, predmets, groups,
prepods
WHERE
predmet_groups.predmet_group_id='".$_GET['g_id']."' and
predmet_groups.prepod_id=prepods.prepod_id and
predmet_groups.predmet_id=predmets.predmet_id and
predmet_groups.group_id=groups.group_id");
$g=mysql_fetch_array($query_group);
echo("<h2 align='center'>Журнал успеваемости '".$_g['group_name']."' класса</h2>
<hr class='clear-contentunit' />
<h2 align='center'>Предмет \"".$_g['predmet_name']."\"</h2>
");
?>

<form name="lform" method="POST">
<table class="statistics">
<tbody>

        <th>Дата выставления оценки</th>

        <tr>

<td style="width:200px;"><input class='dates' type="text" name="data"
id="myOtherInput" />

        <script language="JavaScript">
var o_cal = new tcal ({
    // form name
    'formname': 'lform',
    // input name
    'controlname': 'data'
});
```

```

        // individual template parameters can be modified via the calendar
        variable
        o_cal.a_tpl.yearscroll = false;
        o_cal.a_tpl.weekstart = 1;

        </script>
        </td>

<td style='border:0px;'>
    <input type='submit' value='' name='view_table' style='width: 20px;
height: 20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none;
cursor: pointer;' />
    </td>
</tr>
</tbody></table></form>
    <h3>Пустое поле - текущая дата</h3>

<form name="lform" method="POST">
<table class="statistics">
<tbody>

    <th>Дата домашнего задания</th>

    <tr>

<td style="width:200px;"><input class='dates' type="text" name="data"
id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({
                // form name
                'formname': 'lform',
                // input name
                'controlname': 'data'
            });

            // individual template parameters can be modified via the calendar
            variable
            o_cal.a_tpl.yearscroll = false;
            o_cal.a_tpl.weekstart = 1;

            </script>
            </td>

<td style='border:0px;'>
    <input type='submit' value='' name='view_zadanie' style='width: 20px;
height: 20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none;
cursor: pointer;' />
    </td>
</tr>
</tbody></table></form>

<?
if (isset($_POST['view_zadanie'])) {
echo("
<table class='statistics'>
<th>Домашнее задание</th>
<tr>

```

```

<td style='border:0;'>
<textarea rows='10' cols='55'></textarea>
</td>
</tr>
<tr>
<td style='border:0px;'>
    <input type='submit' value='' name='view_zadanie' style='width: 20px;
height: 20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none;
cursor: pointer;' />Сохранить!
    </td>
</tr>
</table>
");
}

if (isset($_POST['view_table'])) {
if ($_POST['data']=="") {
$dt=date('Y-m-d');
$d = explode("-", $dt);
$v_data="$d[0]$d[1]$d[2]";
} else {
$d = explode("-", $_POST['data']); //раскладываем
$v_data="$d[0]$d[1]$d[2]";
}
?><form method='post'>
<h2>Тема занятия:</h2>
<p>
<textarearows="4" cols="40">Введите тему занятия</textarea>
</p>
<p>
<input type='submit' value='Сохранить' name='go'>
</form>
<?

echo("<h2>Список класса:</h2>
<h3>Выставление оценок на <b>$d[2].$d[1].$d[0]г.</b></h3>
");

?>
<table class='statistics'>
<tr>
<th>#</th>
<th>ФИО</th>
</tr>

<?
$query_stud=mysql_query("SELECT * FROM users WHERE
group_id=".$_g['group_id'].");
$i=0;
$j=1;
While ($st=mysql_fetch_array($query_stud)) {

echo("
<form method='POST' onsubmit='return added();'>
<tr>
");
echo("
<input type='hidden' name='user[$i]' value='".$st['user_id']."'>
<input type='hidden' name='dates' value='$v_data'>
<td class='n'>$j</td>

```

```

<td class='name'>".$st['user_name']. "</td>
");

$query_oc=mysql_query("SELECT * FROM journal WHERE data=$v_data and
predmet_group_id=".$$_GET['g_id']." and user_id=".$st['user_id']. "");
$t=mysql_fetch_array($query_oc);
if ($t!="") {
echo("
<td class='n'><input type='text' class='journal' value='".$t['ocenka']."'
disabled></td>
");
} else {
echo("
<td class='n'><input type='text' class='journal' name='fname[$i]'></td>
");
}
$i=$i+1;
}
echo("
</tr>

");

$j=$j+1;
}
?>

</table>

<input type='submit' value='Сохранить' name='go'>
</form>
<?>?>
<a href='index.php?sect=prepod&page=index'>Выбрать класс</a>
</div>

```

Pages/prepod/view_advert.php

```

<?
$school=mysql_fetch_array(mysql_query("SELECT * FROM schools WHERE
school_id='".$$_SESSION['school_id']."'"));
$predmet_list=mysql_query("SELECT * FROM predmets, prepods WHERE
prepods.prepod_id='".$$_SESSION['user_id']."' and
predmets.predmet_id=prepods.predmet_id");
?>
<div class="column1-unit">
    <h1 align='center'><?echo $_SESSION['user_name'];?></h1>
    <h3>Школа: <?echo $school['school_name'];?></h3>
<br>
<h3>Преподаватель дисциплины:
<?

while($p_list=mysql_fetch_array($predmet_list))
{
echo $p_list['predmet_name'];
}
?>
</h3>
<div class='left_block_p'>
<h1 align='center'>Добавленные объявления</h1>
</div>
<?
if (isset($_GET['delete'])) {
$q_delete=mysql_query("DELETE FROM advert WHERE advert_id='".$_GET['delete']."'");

```

```

if (!$q_delete) echo mysql_error(); else echo ("<p align='center'><font
color='red'>Объявление удалено!!!</font></p>");
}

$my_advert=mysql_query("select * from advert Where
out_id=".$SESSION['user_id']."' ORDER BY advert_id DESC");
if (!$my_advert) echo mysql_error();

$i=1;
while($my_advert_array=mysql_fetch_array($my_advert)){
$mes=mysql_query("select * from groups Where
group_id=".$my_advert_array['in_id']."'");
$mes_user=mysql_fetch_array($mes);
echo("
<div class='message_view'>
<b>Объявление для класса: ".$mes_user['group_name']."'</b><br>
Дата : ".$my_advert_array['advert_date']."'
<div class='message'>
".$my_advert_array['advert']."'
</div>
<p align='right'><a
href='index.php?sect=prepod&page=view_advert&delete=".$my_advert_array['advert_i
d']."'>Удалить</a></p>

</div>
");
$i=$i+1;
}
if ($i==1) echo ("<h2 align='center'>У Вас нет объявлений!</h2><h3>Вы можете
добавить объявление классу перейдя в меню \"Классы\".</h3>")
?>

</div>

Pages/prepod/advert.php
<script language="javascript">
function sendform() {
if (document.mess.text_message.value == "") {
alert('Пожалуйста, введите текст сообщения');
document.mess.text_message.focus();
return false
}
return true;
}
</script>
<?
$school=mysql_fetch_array(mysql_query("SELECT * FROM schools WHERE
school_id='".$SESSION['school_id']."'"));
$predmet_list=mysql_query("SELECT * FROM predmets, prepods WHERE
prepods.prepod_id='".$SESSION['user_id']."' and
predmets.predmet_id=prepods.predmet_id");
?>
<div class="column1-unit">
<h1 align='center'><?echo $SESSION['user_name'];?></h1>
<h3>Школа: <?echo $school['school_name'];?></h3>
<br>
<h3>Преподаватель дисциплины:
<?

```

```

while($p_list=mysql_fetch_array($predmet_list))
{
echo $p_list['predmet_name'];
}
?>
</h3>
<div class='left_block_p'>
<h1 align='center'>Объявления</h1>
</div>

<?
if (isset($_POST['mail_go'])) {
$date=date('Y-m-d H:i:s');
$mess_add=mysql_query("INSERT INTO advert
VALUES('','".$_SESSION['user_id']. "','".$_POST['in_id']. "','".$_POST['text_messa
ge']. "','$date','')");

}

$school=mysql_fetch_array(mysql_query("SELECT * FROM groups WHERE
groups.group_id='".$_SESSION['group_id']."'"));

if (!$mess_add) echo mysql_error(); else echo("<h4 align='center'>Объявление
добавлено!!!<br>

</h4>");

$my_g=mysql_query("SELECT * FROM groups WHERE group_id=".$_GET['g_id']."'");
$my_g_array=mysql_fetch_array($my_g);
echo("
<div class='contactform'>
<form method='post' name='mess' onsubmit='return sendform();'>
    <fieldset>
        <legend>
            &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&Добавить объявление&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
        </legend>
        <p>

            <h3>Объявление для \"".$_my_g_array['group_name']. "\"</h3>

        </p>
        <p>
            <label for='contact_message' class='left'>Текст объявления</label>
<textarea name='text_message' cols='45' rows='5'></textarea>
            <input type='hidden' name='in_id'
value='".$_my_g_array['group_id']."'>
        </p>
        <p>
            <input type='submit' name='mail_go' class='button' value='Отправить'
/>
        </p>
    </fieldset>
</form>
</div>
");
?>
</div>

```

Pages/prepod/index.php

<?

```

$school=mysql_fetch_array(mysql_query("SELECT * FROM schools WHERE
school_id='".$$_SESSION['school_id']."'"));
$predmet_list=mysql_query("SELECT * FROM predmets, prepods WHERE
prepods.prepod_id='".$$_SESSION['user_id']."' and
predmets.predmet_id=prepods.predmet_id");
?>
<div class="column1-unit">
    <h1 align='center'><?echo $_SESSION['user_name'];?></h1>
    <h3>Школа: <?echo $school['school_name'];?></h3>
<br>
<h3>Преподаватель дисциплины:
<?

while($p_list=mysql_fetch_array($predmet_list))
{
echo $p_list['predmet_name'];
}
?>
</h3>

<h1 align='center'>Список классов:</h1>
<h3>Для того чтобы зайти в журнал нужно выбрать класс:</h3>
<table class='statistics'>
<tr>
<th>№</th>
<th style='text-align:left;'>Класс</th>
</tr>
<?
$list_group=mysql_query("SELECT * FROM groups, predmet_groups WHERE
predmet_groups.prepod_id='".$$_SESSION['user_id']."' and
groups.group_id=predmet_groups.group_id");
$i=0;
while($glist=mysql_fetch_array($list_group))
{
$i=$i+1;
echo ( "
<tr>
<td width='10' style='border:0;'>$i</td>
<td style='border:0;'><a
href='index.php?sect=prepod&page=journal&g_id='".$glist['predmet_group_id']."'>".
$glist['group_name']."' класс</a></td>
<td width='100' style='border:0;'><a
href='index.php?sect=prepod&page=advert&g_id='".$glist['group_id']."'
'>Объявление</td>
</tr>
");
}
?>
</table>

</div>

```

Pages/school/rate_school.php

```

<?
$q=mysql_fetch_array(mysql_query("SELECT * FROM schools WHERE
school_id='".$$_SESSION['school_id']."'"));
?>
<div class="column1-unit">
    <h1 align='center'><?echo $q['school_name'];?></h1>
    <hr class="clear-contentunit" />

```

```

<h2>Адрес: <?echo $q['school_adress'];?></h2>
<h2>Телефон: <?echo $q['school_tel'];?></h2>
<hr class="clear-contentunit" />
<h1 align='center'>Рейтинги:</h1>
<table class='statistics'>
<tr>
<td style='border:0;'>Рейтинг по группам:</td>
<td style='border:0px;'>
<form name='rate_s' method='POST'>
    <input type='submit' value='' name='rate_group' style='width: 20px;
height: 20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none;
cursor: pointer;' />
</form>
</td>

<tr>
<tr>
<td style='border:0;'>Рейтинг по студентам:</td>

<td style='border:0px;'>
<form name='rate_s' method='POST'>
    <input type='submit' value='' name='rate_student' style='width: 20px;
height: 20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none;
cursor: pointer;' />
</form>
</td></tr>
</table>
<?
if (isset($_POST['rate_group'])) {
$r_group=mysql_query("SELECT * FROM groups WHERE school_id =
".$_SESSION[school_id].");
if (!$r_group) echo mysql_error();
echo("<table class='statistics'>
<th>Название класса</th>");
while($r_group_array=mysql_fetch_array($r_group)) {
echo("
<tr>
<td>".$_r_group_array['group_name']. "</td>
</tr>
");
}
echo("</table>");
}
if (isset($_POST['rate_student'])) {
echo("Выбран студент!");
}
?>
</div>

```

Pages/school/chetvert.php

```

<script language="JavaScript" src="js/calendar_us.js"></script>
<link rel="stylesheet" href="css/calendar.css">
<?
$q=mysql_fetch_array(mysql_query("SELECT * FROM schools WHERE
school_id='".$_SESSION['school_id']."'"));

?>
<div class="column1-unit">
    <h1 align='center'><?echo $q['school_name'];?></h1>
    <hr class="clear-contentunit" />
<h2>Адрес: <?echo $q['school_adress'];?></h2>
<h2>Телефон: <?echo $q['school_tel'];?></h2>
<hr class="clear-contentunit" />

```

```

<h1 align='center'>Настройки</h1>

<p>1 четверть</p>

<form name="lform" method="POST">
<table class="statistics">
<tbody>

        <th>Дата начала 1 четверти</th>
        <th>Дата окончания 1 четверти</th>
        <tr>

<td style="width:200px;"><input class='dates' type="text" name="data"
id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({
                // form name
                'formname': 'lform',
                // input name
                'controlname': 'data'
            });

            // individual template parameters can be modified via the calendar
            variable
            o_cal.a_tpl.yearscroll = false;
            o_cal.a_tpl.weekstart = 1;

        </script>
        </td>
        <td style="width:200px;"><input class='dates'
type="text" name="data2" id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({
                // form name
                'formname': 'lform',
                // input name
                'controlname': 'data2'
            });

            // individual template parameters can be modified via the calendar
            variable
            o_cal.a_tpl.yearscroll = false;
            o_cal.a_tpl.weekstart = 1;

        </script>
        </td>

<td style='border:0px;'>
        <input type='submit' value='' name='view' style='width: 20px; height:
20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none; cursor:
pointer;' />
        </td>
</tr>
</tbody></table></form>

<p>2 четверть</p>

<form name="lform2" method="POST">
<table class="statistics">
<tbody>

```

```

        <th>Дата начала 2 четверти</th>
        <th>Дата окончания 2 четверти</th>
    <tr>

<td style="width:200px;"><input class='dates' type="text" name="data"
id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({
                // form name
                'formname': 'lform',
                // input name
                'controlname': 'data'
            });

            // individual template parameters can be modified via the calendar
variable
            o_cal.a_tpl.yearsroll = false;
            o_cal.a_tpl.weekstart = 1;

        </script>
        </td>
        <td style="width:200px;"><input class='dates'
type="text" name="data2" id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({
                // form name
                'formname': 'lform',
                // input name
                'controlname': 'data2'
            });

            // individual template parameters can be modified via the calendar
variable
            o_cal.a_tpl.yearsroll = false;
            o_cal.a_tpl.weekstart = 1;

        </script>
        </td>

<td style='border:0px;'>
    <input type='submit' value='' name='chetvert1' style='width: 20px; height:
20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none; cursor:
pointer;' />
    </td>
</tr>
</tbody></table></form>

<p>3 четверть</p>

<form name="lform2" method="POST">
<table class="statistics">
<tbody>

        <th>Дата начала 3 четверти</th>
        <th>Дата окончания 3 четверти</th>
    <tr>

<td style="width:200px;"><input class='dates' type="text" name="data"
id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({

```

```

        // form name
        'formname': 'lform',
        // input name
        'controlname': 'data'
    });

    // individual template parameters can be modified via the calendar
    variable
    o_cal.a_tpl.yearscroll = false;
    o_cal.a_tpl.weekstart = 1;

    </script>
        </td>
        <td style="width:200px;"><input class='dates'
type="text" name="data2" id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({
                // form name
                'formname': 'lform',
                // input name
                'controlname': 'data2'
            });

            // individual template parameters can be modified via the calendar
            variable
            o_cal.a_tpl.yearscroll = false;
            o_cal.a_tpl.weekstart = 1;

            </script>
        </td>

<td style='border:0px;'>
    <input type='submit' value='' name='chetvert1' style='width: 20px; height:
20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none; cursor:
pointer;' />
</td>
</tr>
</tbody></table></form>

<p>4 четверть</p>

<form name="lform2" method="POST">
<table class="statistics">
<tbody>

        <th>Дата начала 4 четверти</th>
        <th>Дата окончания 4 четверти</th>
        <tr>

<td style="width:200px;"><input class='dates' type="text" name="data"
id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({
                // form name
                'formname': 'lform',
                // input name
                'controlname': 'data'
            });

            // individual template parameters can be modified via the calendar
            variable

```

```

        o_cal.a_tpl.yearscroll = false;
        o_cal.a_tpl.weekstart = 1;

</script>
</td>
<td style="width:200px;"><input class='dates'
type="text" name="data2" id="myOtherInput" />
<script language="JavaScript">
    var o_cal = new tcal ({
        // form name
        'formname': 'lform',
        // input name
        'controlname': 'data2'
    });

    // individual template parameters can be modified via the calendar
variable
    o_cal.a_tpl.yearscroll = false;
    o_cal.a_tpl.weekstart = 1;

</script>
</td>

<td style='border:0px;'>
    <input type='submit' value='' name='chetvert1' style='width: 20px; height:
20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none; cursor:
pointer;' />
</td>
</tr>
</tbody></table></form>

</div>

```

Pages/school/search.php

```

<?
$q=mysql_fetch_array(mysql_query("SELECT * FROM schools WHERE
school_id='".$$_SESSION['school_id']."'"));

?>
<div class="column1-unit">
    <h1 align='center'><?echo $q['school_name'];?></h1>
    <hr class="clear-contentunit" />
<h2>Адрес: <?echo $q['school_adress'];?></h2>
<h2>Телефон: <?echo $q['school_tel'];?></h2>
<hr class="clear-contentunit" />
<h1 align='center'>Поиск ученика в БД:</h1>
<table class='statistics'>
<th>Введите признаки для поиска</th>
<tr>
<form method='POST'>
<td style='border:0;'>
<input type='text' name='find' width='300'>
</td>
<td style='border:0;'>
<input type='submit' name='find_ok' value='Найти'>
</td>
</tr>
</table>
<?
if (isset($_POST['find_ok'])) {

$find_query=mysql_query("SELECT *

```

```

FROM
groups
INNER JOIN users ON users.group_id = groups.group_id
INNER JOIN schools ON groups.school_id = schools.school_id
WHERE
schools.school_id = 66 AND
users.user_name LIKE '%"$_POST['find'].%"';
if (!$find_query) echo mysql_error();
echo("<table class='statistics'>
<th>№</th>
<th>ФИО</th>
<th>Класс</th>");
$i=1;
while($find_array=mysql_fetch_array($find_query)){
echo("
<tr><td>$i</td>
<td>".$_find_array['user_name'].</td>
");
echo("
<td>".$_find_array['group_name'].</td>
</tr>

");
$i=$i+1;
}
echo("</table>");
if ($i==1) echo("<h3>К сожалению ничего не найдено!</h3>");
else
{
$i=$i-1;
echo("Найдено <b>$i</b> записей...");
}
}
?>
</div>

```

Pages/school/prepod.php

```

<script>
function confirmDelete() {
    if (confirm("Вы подтверждаете удаление?")) {
        return true;
    } else {
        return false;
    }
}
</script>
<?
if (isset($_POST['del_prepod'])) {
    $delete_prepod=mysql_query("DELETE FROM prepods WHERE
    prepod_id='".$_POST['del_value']."'");
    if (!$delete_prepod) echo ("Ошибка при удалении... Попробуйте еще
    раз позже!!!");
}

function rndstr($length)
{
    $return = $chars = null;
    $chars .= 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789';
    for($i = 0, $sl = strlen($chars) - 1; $i <= $length; $i++)
    $return .= $chars[rand(0, $sl)];
    return $return;
}

```

```

}

function test_added() {
    $_SESSION['err_add_prepod']="";
    if (empty($_POST['add_prepod_name']))
    {
        $_SESSION['err_add_prepod']="Не введено ФИО преподавателя.";
        return (false);
    }
    return (true);
}

if (isset($_POST['add_prepod_submit'])) {

    if (test_added()) {
        $t=true;
        while ($t):
            $code_access=rndstr(7);
            $res=mysql_query("SELECT * FROM prepods WHERE
prepod_psw='".$code_access.'"");
            $rows=mysql_num_rows($res);
            if ($rows==0) $t=false;
            endwhile;

            $q=mysql_query("INSERT INTO prepods
VALUES('','".$_SESSION['school_id']. "','".$_POST['add_prepod_name']. "','".$_POST[
'add_predmet_name']. "','$code_access')");
            if (!$q) echo ("Ошибка при добавлении... Попробуйте еще раз
позже!!!");

        }
    }

    $q=mysql_fetch_array(mysql_query("SELECT * FROM schools WHERE
school_id='".$_SESSION['school_id']. '""));

    ?>
    <div class="column1-unit">
        <h1 align='center'><?echo $q['school_name'];?></h1>
        <hr class="clear-contentunit" />
        <h2>Адрес: <?echo $q['school_adress'];?></h2>
        <h2>Телефон: <?echo $q['school_tel'];?></h2>
        <hr class="clear-contentunit" />
        <h1 align='center'>Преподаватели:</h1>

    <?
        $count_prepod=mysql_query("SELECT * FROM prepods, predmets WHERE
prepods.school_id='".$_SESSION['school_id']. "' and
prepods.predmet_id=predmets.predmet_id");
        $count_p=mysql_num_rows($count_prepod);
        if ($count_p>0) {
            ?>
            <table class='statistics'>
            <tr>
            <th>ФИО преподавателя</th>
            <th>Дисциплина</th>
            <th>Код доступа</th>
            </tr>

            <?

```

```

        while($mas=mysql_fetch_array($count_prepod))
        {
            ?>
            <tr>
            <td><?echo $mas['prepod_name'];?></td>
            <td><?echo $mas['predmet_name'];?></td>
            <td><?echo $mas['prepod_psw'];?></td>
            <form method='POST' onSubmit="return confirmDelete()">
            <td style='border:0;'><input type='submit' value='' name='del_prepod'
            style='width: 10px; height: 10px; background: url(img/btn_del2.png) 0 0 no-
            repeat; border: none; cursor: pointer;' alt='Удалить преподавателя' />
            <input type='hidden' name='del_value' value='<?echo $mas['prepod_id'];?>'></td>
            </form>
            </tr>
            <?><?>

        </table>
        <?> else echo("<h2 align='center'>Преподаватели отсутствуют</h2>");?>
        <hr class="clear-contentunit" />
        <br>
        <h3>Добавление нового преподавателя:</h3>
        <form name='add_prepod' method='POST'>
        <table class="statistics" align='left'>
        <tr>
        <td width="30" style='border:0;'>ФИО:</td>
        <td style='border:0;'><input type='text' class='c1' name='add_prepod_name'></td>

        </tr>
        <td width="30" style='border:0;'>Дисциплина:</td>
        <? $predmet_query = mysql_query("SELECT * FROM predmets"); ?>
        <td style='border:0;'>
        <SELECT name="add_predmet_name">
            <? while($predmet_mas = mysql_fetch_array($predmet_query))
            {
                echo "<option
value='\".$predmet_mas['predmet_id'].\"'>\".$predmet_mas['predmet_name'].\"</option>
\";
                }?>
        </SELECT>
        </td>
        <td width="50" style='border:0;'><input type="submit" name="add_prepod_submit"
        class="button" value="Добавить"/></td>
        <tr>
        </tr>
        </table>
        </form>
        <p align='right'><a href="javascript: history.go(-1)">Назад</a></p>
        <hr class="clear-contentunit" />
        </div>

```

Pages/school/class.php

```

<script>
functionconfirmDelete() {
    if (confirm("Вы действительно хотите удалить класс и находящихся в нем
студентов?")) {
        return true;
    } else {
        return false;
    }
}
</script>

```

```

<?
if (isset($_POST['del_class'])) {
    $delete_class=mysql_query("DELETE FROM groups WHERE
group_id='".$_$_POST['del_value']."'");
    if (!$delete_class) echo ("Ошибка при удалении... Попробуйте еще раз
позже!!!");
}

function test_added() {
    $_SESSION['err_add_class']="";
    if (empty($_POST['add_class_name']))
    {
        $_SESSION['err_add_class']="Не введено название класса.";
        return (false);
    }
    return (true);
}
if (isset($_POST['add_class_submit'])) {
    if (test_added()) {
        $q=mysql_query("INSERT INTO groups
VALUES ('', '".$_$_SESSION['school_id'].'', '".$_$_POST['add_class_name'].'', '".$_$_POST['
add_prepod_name'].')");
        if (!$q) echo ("Ошибка при добавлении... Попробуйте еще раз
позже!!!");
    }
}

$q=mysql_fetch_array(mysql_query("SELECT * FROM schools WHERE
school_id='".$_$_SESSION['school_id']."'"));

?>
<div class="column1-unit">
    <h1 align='center'><?echo $q['school_name'];?></h1>
    <hr class="clear-contentunit" />
<h2>Адрес: <?echo $q['school_adress'];?></h2>
<h2>Телефон: <?echo $q['school_tel'];?></h2>
<hr class="clear-contentunit" />
<h1 align='center'>Классы:</h1>
<table class="statistics" align='left'>
<tr>
<th>Название класса</th>
<th>Классный руководитель</th>
</tr>
<?
    $count_class=mysql_query("SELECT * FROM groups,prepods WHERE
prepods.school_id='".$_$_SESSION['school_id']."' and
groups.prepod_id=prepods.prepod_id");
    $i=1;
    while($mas=mysql_fetch_array($count_class))
    {

?>
<form name="del_class" method="POST" onSubmit="return confirmDelete()">
<tr>
<td width="100"><a href='index.php?sect=schools&page=stud&g_id=<?echo
$mas['group_id'];?>'><?echo $mas['group_name'];?> класс</a></td>
<td width="270"><?echo $mas['prepod_name'];?></td>
<td width="20" style='border:0;'>

```



```

$count_class=mysql_num_rows(mysql_query("SELECT * FROM groups WHERE
school_id='".$$_SESSION['school_id']."'"));
$count_user=mysql_num_rows(mysql_query("SELECT * FROM users, groups WHERE
school_id='".$$_SESSION['school_id']."' and users.group_id=groups.group_id"));
$count_prepod=mysql_num_rows(mysql_query("SELECT * FROM prepos WHERE
school_id='".$$_SESSION['school_id']."'"));
$count_predmet=mysql_num_rows(mysql_query("SELECT * FROM predmets, groups,
predmet_groups WHERE school_id='".$$_SESSION['school_id']."' and
predmets.predmet_id=predmet_groups.predmet_id and
groups.group_id=predmet_groups.group_id"));
??
<p>Количество классов: <?echo $count_class;?></p>
<p>Количество учащихся: <?echo $count_user;?></p>
<p>Количество преподавателей: <?echo $count_prepod;?></p>
<p>Количество предметов: <?echo $count_predmet;?></p>
</div>

```

Pages/school/stud.php

```

<script type="text/javascript" src="js/linkedselect.js"></script>
<script>
function confirmDelete() {
    if (confirm("Вы подтверждаете удаление?")) {
        return true;
    } else {
        return false;
    }
}
</script>
<?
function test_added_predmet() {
    $_SESSION['err_add_prepod']="";
    if (empty($_POST['add_prep']))
    {
        $_SESSION['err_add_prepod']="Не введено имя преподавателя";
        return (false);
    }
    if (empty($_POST['add_pred']))
    {
        $_SESSION['err_add_prepod']="Не введено названия дисциплины";
        return (false);
    }
    return (true);
}
if (isset($_POST['add_predmet_submit'])) {

    if (test_added_predmet()) {

        $q=mysql_query("INSERT INTO predmet_groups
VALUES('','".$_GET['g_id']."'','".$_POST['add_pred']."'".$_POST['add_prep']."'')");
        if (!$q) echo ("Ошибка при добавлении... Попробуйте еще раз
позже!!!");
        else $good = "Дисциплина добавлена!!!";
    }

    if (isset($_POST['del_record'])) {
        $delete_user=mysql_query("DELETE FROM users WHERE
user_id='".$_POST['del_value']."'");
        if (!$delete_user) echo ("Ошибка при удалении... Попробуйте еще раз
позже!!!");
    }
}

```

```

}
if (isset($_POST['del_pred'])) {
    $delete_user=mysql_query("DELETE FROM predmet_groups WHERE
    predmet_group_id='".$_POST['del_value']."'");
    if (!$delete_user) echo ("Ошибка при удалении... Попробуйте еще раз
    позже!!!");
}

function rndstr($length)
{
    $return = $chars = null;
    $chars .= 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789';
    for($i = 0, $sl = strlen($chars) - 1; $i <= $length; $i++)
    $return .= $chars[rand(0, $sl)];
    return $return;
}
function test_added() {
    $_SESSION['err_add_user']="";
    if (empty($_POST['user_name_add']))
    {
        $_SESSION['err_add_user']="Не введено имя учащегося.";
        return (false);
    }
    return (true);
}

if (isset($_POST['add_user_submit'])) {

    if (test_added()) {
        $t=true;
        while ($t):
            $code_access=rndstr(7);
            $res=mysql_query("SELECT * FROM users WHERE
user_psw='".$_code_access.'");
            $rows=mysql_num_rows($res);
            if ($rows==0) $t=false;
        endwhile;

        $q=mysql_query("INSERT INTO users
VALUES('','".$_GET['g_id']. "','".$_POST['user_name_add']. "','$code_access')");
        if (!$q) echo ("Ошибка при добавлении... Попробуйте еще раз
        позже!!!");
        else $good = "Ученик ".$_POST['user_name_add']. " добавлен в класс";
    }
}

$q=mysql_fetch_array(mysql_query("SELECT * FROM schools WHERE
school_id='".$_SESSION['school_id']."'"));

?>
<div class="column1-unit">
    <h1 align='center'><?echo $q['school_name'];?></h1>
    <hr class="clear-contentunit" />
<h2>Адрес: <?echo $q['school_adress'];?></h2>
<h2>Телефон: <?echo $q['school_tel'];?></h2>
<hr class="clear-contentunit" />
<? $class_n=mysql_fetch_array(mysql_query("SELECT * FROM groups, prepods WHERE
groups.group_id='".$_GET['g_id']."' and prepods.prepod_id=groups.prepod_id"));?>
<h1 align='center'><? echo $class_n['group_name'];?> класс:</h1>

```

```

<h3 align='right'>Классный руководитель: <b><? echo
$class_n['prepod_name']?><b></h3>

<h2>Список учеников:</h2>
<table class="statistics" align='left'>
<tr>
<th>№</th>
<th>ФИО ученика</th>
<th>Пароль</th>
</tr>
<?
    $class_user=mysql_query("SELECT * FROM users WHERE
group_id=".$$_GET['g_id']." ORDER BY user_name");
    $i=1;
    while($mas=mysql_fetch_array($class_user))
    {
?>
<form name="del_user" method="POST" onSubmit="return confirmDelete()">
<tr>
<td width="20"><?echo $i;?></td>
<td width="300"><?echo $mas['user_name'];?></a></td>
<td width="50"><?echo $mas['user_psw'];?></td>
<td width="20" style='border:0;'>
<input type='submit' value='' name='del_record' style='width: 20px; height:
20px; background: url(img/btn_del2.png) 0 0 no-repeat; border: none; cursor:
pointer;' alt='Удалить ученика' />
<input type='hidden' name='del_value' value='<?echo $mas['user_id'];?>'>
</td>
</tr>
</form>
<?
$i=$i+1;
}?>
</table>
<?
if ($i==1) echo("<palgin='center'>В классе отсутствуют учащиеся! Для добавления
используйте форму ниже.</p>");
if (($_SESSION['err_add_user'])!="") echo ("<center><h4><font color='red'>
".$_SESSION['err_add_user'].</font></h4></center>");

?>
<h3>Добавление нового ученика в класс:</h3>
<form name='add_user' method='POST'>
<table class="statistics" align='left'>
<tr>
<td width="30" style='border:0;'>ФИО:</td>
<td style='border:0;'><input type='text' class='c1' name='user_name_add'></td>
<td width="50" style='border:0;'><input type="submit" name="add_user_submit"
class="button" value="Добавить"/></td>
</tr>
</table>
</form>
<p align='right'><a href="javascript: history.go(-1)">Назад</a> | <a
href="index.php?sect=schools&page=class">Все классы</a></p>
<hr class="clear-contentunit" />

<h2>Учебные дисциплины:</h2>
<table class='statistics'>
<tr>
<th>№</th>
<th>Дисциплина</th>
<th>Преподаватель</th>

```

```

        </tr>
        <?
        $predmet_groups=mysql_query("SELECT * FROM predmet_groups, prepods,
predmets WHERE predmet_groups.group_id=".$$_GET['g_id']." AND prepods.prepod_id =
predmet_groups.prepod_id
AND predmets.predmet_id = predmet_groups.predmet_id");
        $i=1;
        while($mas2=mysql_fetch_array($predmet_groups))
        {
?>
        <form name="del_pred" method="POST" onSubmit="return confirmDelete()">
        <tr>
        <td>1</td>
        <td><? echo $mas2['predmet_name'];?></td>
        <td><? echo $mas2['prepod_name'];?></td>
        <td width="20" style='border:0;'>
        <input type='submit' value='' name='del_pred' style='width: 20px; height: 20px;
background: url(img/btn_del2.png) 0 0 no-repeat; border: none; cursor:
pointer;' alt='Удалить ученика' />
        <input type='hidden' name='del_value' value='<?echo
$mas2['predmet_group_id'];?>'>

        </td>
        </tr>
        </form>
        <?}?>
</table>

<h3>Добавление новой дисциплины:</h3>
<form name='add_predmet' method='POST'>
<table class="statistics" align='left'>
<tr>
<td width="30" style='border:0;'>Дисциплина:</td>
<td style='border:0;'>

<!-- Первый (главный) список (изначально заполнен вручную) -->
<select id="List1" name='add_pred'>
<?
$predmety=mysql_query("SELECT * FROM predmets ORDER BY predmet_name");

        while($mas_predmet=mysql_fetch_array($predmety))
        {
?>
        <option value="<?echo $mas_predmet['predmet_id'];?>"><? echo
$mas_predmet['predmet_name'];?></option>
        <?}?>
        </select>

<!-- Подчиненный список 1 (изначально пуст) -->
</td>
</tr>
<tr>
<td style='border:0;'>Преподаватель</td>
<td style='border:0;'>
<select id="List2" name="add_prep"></select>

<!-- Подчиненный список 2 (изначально пуст) -->

<scripttype="text/javascript">

```

```

// Создаем новый объект связанных списков
varsyncList1 = newsyncList;

// Определяем значения подчиненных списков (2 и 3 селектов)
syncList1.dataList = {

/* Определяем элементы второго списка в зависимости
от выбранного значения в первом списке */
<?
$class_user=mysql_query("SELECT * FROM predmets");

    while($mas=mysql_fetch_array($class_user))
    {
?>
'<?echo $mas['predmet_id'];?>':{ //
<?
    $class_user2=mysql_query("SELECT * FROM prepods WHERE
predmet_id=".$mas['predmet_id']." and school_id='".$$_SESSION['school_id']."'");

    while($mas1=mysql_fetch_array($class_user2))
    {
?>
'<?echo $mas1['prepod_id'];?>': '<?echo $mas1['prepod_name'];?>',
<?}?>

    },
<?}?>
};

// Включаем синхронизацию связанных списков
syncList1.sync("List1", "List2");
</script>

</td>

</tr>
<tr>
<td style='border:0;'></td>
<td width="50" style='border:0;'><input type="submit" name="add_predmet_submit"
class="button" value="Добавить"/></td>
</tr>
</table>
</form>
<p align='right'><a href="javascript: history.go(-1)">Назад</a> | <a
href="index.php?sect=schools&page=class">Все классы</a></p>
<hr class="clear-contentunit" />
</div>

```

Pages/student/rate.php

```

<?
$school=mysql_fetch_array(mysql_query("SELECT * FROM groups WHERE
groups.group_id='".$$_SESSION['group_id']."'"));

?>
<div class="column1-unit">
<div class='left_block_p'>
<h2 align='center'><?echo $_SESSION['user_name']." ". $school['group_name']."'
класс";?></h2>

```

```

</div>
<h2 align='center'>Мой рейтинг</h2>
<?
$col_all=mysql_query("SELECT
COUNT(*)
FROM
journal
INNER JOIN users ON journal.user_id = users.user_id
WHERE
users.group_id = ".$SESSION['group_id'].");
$t_all=mysql_fetch_row($col_all);
$user1=mysql_query("SELECT * FROM users WHERE group_id =
".$SESSION['group_id'].");

echo("<table class='statistics'>
<th>№</th>
<th>ФИО</th>
<th>Всего оценок</th>
<th>Средний бал</th>");
$i=1;
while($user1_res=mysql_fetch_array($user1)){
$user2=mysql_query("SELECT SUM(ocenka) FROM journal WHERE user_id =
".$user1_res['user_id'].");
$user_fetch = mysql_result($user2,'0');
$user3=mysql_query("SELECT count(*) FROM journal WHERE user_id =
".$user1_res['user_id'].");
$user_count_o=mysql_fetch_row($user3);
$av=$user_fetch/$user_count_o[0];
$av=round($av, 1);

echo("
<tr><td style='border:0;'>$i</td><td>
".$user1_res['user_name'].</td>
<td>
".$user_count_o[0].</td>
<td>
".$av.</td>
</tr>
");
$i=$i+1;
}
echo("</table>");
?>
<p>Количество полученных оценок</p>


</div>
<img src='123.php'>

```

Pages/student/myprepods.php

```

<?
$school=mysql_fetch_array(mysql_query("SELECT * FROM groups WHERE
groups.group_id='".$SESSION['group_id']."'"));

```

```

?>
<div class="column1-unit">
<div class='left_block_p'>
<h2 align='center'><?echo $_SESSION['user_name']." ". $school['group_name']."
класс";?></h2>
</div>
<?
echo("
<h2 align='center'>Мои преподаватели</h2>
");
$myprepods=mysql_query("SELECT
*

FROM
prepods
INNER JOIN predmet_groups ON predmet_groups.prepod_id = prepods.prepod_id
INNER JOIN predmets ON predmets.predmet_id = predmet_groups.predmet_id
WHERE predmet_groups.group_id=".$_SESSION['group_id']."'");

echo("
<table class='statistics'>");
while($myprepods_array=mysql_fetch_array($myprepods))
{
echo("<tr>
<td width='20' style='border:0;'><b>
".$myprepods_array['predmet_name']."</b>
</td>
<td style='border-top:0; border-left:0; border-right:0; padding:5px 0 5px
0;'>".$myprepods_array['prepod_name']."</td>
</tr>");
}
echo("</table>
");
?>
</div>

```

Pages/student/index.php

```

<script language="JavaScript" src="js/calendar_us.js"></script>
<link rel="stylesheet" href="css/calendar.css">
<?
$school=mysql_fetch_array(mysql_query("SELECT * FROM groups WHERE
groups.group_id='".$_SESSION['group_id']."'"));

?>

<div class="column1-unit">
<div class='left_block_p'>
<h1 align='center'>Мой дневник</h1>
<h2 align='center'><?echo $_SESSION['user_name']." ". $school['group_name']."
класс";?></h2>
</div>

<form name="lform" method="POST">
<tableclass="statistics">
<tbody>

        Выберите даты для простомтра дневника

        <th>От</th>
        <th>До</th>
        <tr>

```

```

<td style="width:200px;"><input class='dates' type="text" name="data"
id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({
                // form name
                'formname': 'lform',
                // input name
                'controlname': 'data'
            });

            // individual template parameters can be modified via the calendar
            variable
            o_cal.a_tpl.yearscroll = false;
            o_cal.a_tpl.weekstart = 1;

        </script>
        </td>
        <td style="width:200px;"><input class='dates'
type="text" name="data2" id="myOtherInput" />
        <script language="JavaScript">
            var o_cal = new tcal ({
                // form name
                'formname': 'lform',
                // input name
                'controlname': 'data2'
            });

            // individual template parameters can be modified via the calendar
            variable
            o_cal.a_tpl.yearscroll = false;
            o_cal.a_tpl.weekstart = 1;

        </script>
        </td>

<td style='border:0px;'>
    <input type='submit' value='' name='view' style='width: 20px; height:
20px; background: url(img/btn_ok.png) 0 0 no-repeat; border: none; cursor:
pointer;' />
    </td>
</tr>
</tbody></table></form>
<?
if (isset($_POST['view'])) {

$d1 = explode("-", $_POST['data']); //раскладываем
$v_data1="$d1[0]$d1[1]$d1[2]";
$d2 = explode("-", $_POST['data2']); //раскладываем
$v_data2="$d2[0]$d2[1]$d2[2]";

$o_query=mysql_query("SELECT * FROM predmet_groups, prepods, predmets WHERE
predmet_groups.group_id='".$_SESSION['group_id']."' AND prepods.prepod_id =
predmet_groups.prepod_id
AND predmets.predmet_id = predmet_groups.predmet_id");
?>
<table class='statistics'>
<th>Предметы</th>
<?
$data_query=mysql_query("SELECT * FROM journal WHERE data BETWEEN '$v_data1' and
'$v_data2' GROUP BY data HAVING COUNT(*) >= 1");
while($datas=mysql_fetch_array($data_query))
{

```

```

$dat = explode("-", $datas['data']);
echo("<th>". $dat[2]. ".". $dat[1]. "</th>");
}
while($mas=mysql_fetch_array($o_query))
{
?>

<tr>
<td><?echo $mas['predmet_name']?></td>
<?
$data_query2=mysql_query("SELECT * FROM journal WHERE
predmet_group_id=".$mas['predmet_group_id']. " and data BETWEEN '$v_data1' and
'$v_data2' GROUP BY data HAVING COUNT(*) >= 1");
while($datas2=mysql_fetch_array($data_query2))
{

?>
<td class='n'><? echo $datas2['ocenka'];?></td>
<?

}??>
</tr>

<?}?>
</table>
<?
}
?>

</div>

```

Pages/student/add_mail.php

```

<script language="javascript">
function sendform() {
if (document.mess.text_message.value == "") {
alert('Пожалуйста, введите текст сообщения');
document.mess.text_message.focus();
return false
}
return true;
}
</script>
<?
if (isset($_POST['mail_go'])) {
$date=date('Y-m-d H:i:s');
$mess_add=mysql_query("INSERT INTO message
VALUES('','.$_SESSION['user_id'].','.$_POST['in_id'].','.$_POST['text_messa
ge'].','.$date','')");

}

$school=mysql_fetch_array(mysql_query("SELECT * FROM groups WHERE
groups.group_id='".$_SESSION['group_id']."'"));

?>
<div class="column1-unit">
<div class='left_block_p'>
<h2 align='center'><?echo $_SESSION['user_name']. " ". $school['group_name']. "
класс";?></h2>
</div>

```

[illegible]

Pages/student/mail.php

```

$mymess=mysql_query("select * from message Where in_id=".$_SESSION['user_id'].
ORDER BY message_id DESC");
if (!$mymess) echo mysql_error();

$i=1;
while($mymess_array=mysql_fetch_array($mymess)){
$mes=mysql_query("select * from users Where
user_id=".$_mymess_array['out_id'].");
$mes_user=mysql_fetch_array($mes);
echo (
<div class='message_view'>
<b>#<i> $i Сообщение от:<i> ". $mes_user['user_name']. "</i></b><br>
Дата : ". $mymess_array['message_date'].
<div class='message'>
". $mymess_array['message'].
</div>
<p align='right'><a
href='index.php?sect=student&page=add_mail&user_id=".$_mes_user['user_id']. "'
>Написать сообщение</a> |
<a
href='index.php?sect=student&page=mail&delete=".$_mymess_array['message_id']. "'>у
далить</a></p>

</div>
");
$i=$i+1;
}

if ($i==1) echo ("<h4 align='center'>у вас нет входящих сообщений!!!</h4>");
echo("<h3 style='margin-left:350px;'>Всего сообщений : ".($i-1). "</h3>");
?>
</div>

```

Pages/student/myclass.php

```

<script language="javascript">
function sendform() {
if (document.mess.text_message.value == "") {
alert('Пожалуйста, введите текст сообщения');
document.mess.text_message.focus();
return false
}
return true;
}
</script>
<?
$school=mysql_fetch_array(mysql_query("SELECT * FROM groups WHERE
groups.group_id=".$_SESSION['group_id']. "));

?>
<div class="column1-unit">
<div class='left_block_p'>
<h2 align='center'><?echo $_SESSION['user_name']. " ". $school['group_name']. "
класс";?></h2>
</div>
<?

echo (
<h2 align='center'>Мой класс</h2>
");

```


ПРИЛОЖЕНИЕ Б

презентация