МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего профессионального образования

**«Дальневосточный федеральный университет»**

|  |  |
| --- | --- |
|  | **ШКОЛА ЕСТЕСТВЕННЫХ НАУК** |

**кафедра информатики, математического и компьютерного моделирования**

|  |
| --- |
| Реализация системы контроля успеваемости  на мастер-классах по программированию ДВФУ |

**ПРАКТИЧЕСКАЯ РАБОТА**

по дисциплине Проектный практикум

|  |
| --- |
| Студентов группы Б8419а  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Шморгун К.В  (подпись)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Машнев М.Б.  (подпись)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Харченко Е.А.  (подпись)  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г. |
| Руководитель  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Кленина Н.В.  (подпись)  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г |
| Защищено с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Кленина Н.В.  (подпись) «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г. |
|  |

**г. Владивосток**

**2015**

Оглавление

[Аннотация 3](#_Toc437946200)

[1. Введение 3](#_Toc437946201)

[1.1. Глоссарий 4](#_Toc437946202)

[1.2. Описание предметной области 4](#_Toc437946203)

[1.3. Существующие проблемы 6](#_Toc437946204)

[1.4. Обзор существующих решений 9](#_Toc437946205)

[1.5. Неформальная постановка задачи 10](#_Toc437946206)

[2. Требования к окружению 11](#_Toc437946207)

[2.1. Требования к аппаратному обеспечению 11](#_Toc437946208)

[2.2. Требования к программному обеспечению 11](#_Toc437946209)

[2.3. Требования к пользователям 11](#_Toc437946210)

[2.4. Функциональные требования 12](#_Toc437946211)

[3. Спецификация данных 14](#_Toc437946212)

[3.1. Диаграмма базы данных 14](#_Toc437946213)

[3.2. Описание сущностей 15](#_Toc437946214)

[3.2.1. Сущность «Пользователь» 15](#_Toc437946215)

[3.2.2. Сущность «Участники» 15](#_Toc437946216)

[3.2.3. Сущность «Преподаватель» 16](#_Toc437946217)

[3.2.4. Сущность «Группы» 17](#_Toc437946218)

[3.2.5. Сущность «Занятия» 17](#_Toc437946219)

[3.2.6. Сущность «Статус группы» 18](#_Toc437946220)

[3.2.7. Сущность «Участник/Оценка» 19](#_Toc437946221)

[4. Проект 20](#_Toc437946222)

[4.1. Средства реализации 20](#_Toc437946223)

[4.2. Структуры данных 20](#_Toc437946224)

[4.2.1. Сущность «Статус группы» 20](#_Toc437946225)

[4.2.2. Сущность «Участники» 20](#_Toc437946226)

[4.2.3. Сущность «Преподаватель» 21](#_Toc437946227)

[4.2.4. Сущность «Группы» 21](#_Toc437946228)

[4.2.5. Сущность «Занятия» 22](#_Toc437946229)

[4.2.6. Сущность «Статус группы» 22](#_Toc437946230)

[4.2.7. Сущность «Участник/Оценка» 23](#_Toc437946231)

[4.3. Требование к интерфейсу 24](#_Toc437946232)

[4.4. Диаграмма состояний 25](#_Toc437946233)

[4.5. Проект интерфейса 27](#_Toc437946234)

[5. Внесенные изменения 32](#_Toc437946235)

[Заключение 33](#_Toc437946236)

[Список литературы 34](#_Toc437946237)

Аннотация

Данная работа посвящена созданию онлайн-системы контроля обучения на мастер-классах по программированию, организованных кафедрой информатики, математического и компьютерного моделирования ДВФУ. Система должна автоматизировать организацию учебного процесса мастер-классов. В ходе исследования были выявлены существующие проблемы и разработан начальный интерфейс сайта.

1. Введение

Во все времена одаренные люди, проявляющие интерес к науке, высоко ценились обществом, ведь они были двигателями прогресса. Зачастую, для развития своих способностей, они отправлялись в специализированные заведения, в стенах которых получали необходимые знания. Но и общество не оставалось равнодушным. Со своей стороны, оно всегда старалось идти навстречу таким людям и создавать благоприятные условия для комфортного обучения.

Древнегреческий ученый Архимед, воспитанный в семье математика и астронома Фидия, с детства питал любовь к точным наукам. Наибольший интерес Архимед проявлял к таким предметам, как механика, астрономия и математика. Но семья понимала, что в родном городе у мальчика не было возможности получить достойное образование. Поэтому отец отправил юное дарование на обучение в научный и культурный центр того времени – Александрию Египетскую. Именно там он познакомился с трудами Демокрита, Евдокса и других великих греческих математиков, которые вдохновили его и дали толчок будущим открытиям. [[1]](#arhimed)

Другой, не менее известный ученый, Алан Мэтисон Тьюринг, английский математик, логик и криптограф, оказавший существенное влияние на развитие информатики, обучался совсем иначе. Как только мальчик пошел в начальную школу, директор сразу же отметил его необычайную одаренность и любовь к математике. В 13 лет Тьюринг поступил в частную школу Шерборн, где его увлечение математикой не нашло особой поддержки среди учителей, так как больше внимания уделялось гуманитарным наукам. Несмотря на то, что в школе не преподавали даже основ математического анализа, Тьюринг самостоятельно решал сложные технические задачи, изучал работы Эйнштейна и других известных ученых. Таким образом, только благодаря своим собственным силам, Алан Тьюринг прославился на весь мир. [[2]](#wikipedia)

Исходя из этого, можно понять, что общество не всегда поддерживает стремление одаренных людей к развитию. Без поддержки одаренность пропадает, только единицы достигают желаемого успеха, остальным же, полагаясь только на собственные силы, приходится труднее. В связи с этим очень важно, чтобы желания людей и общества совпали.

* 1. Глоссарий

*Рейтинг* – индивидуальный числовой показатель, характеризующий успеваемость конкретного участника.

*Ступень мастер класса* – уровень знаний участника в области программирования, отражающийся в принадлежности его к определенной группе.

*CATS* – автоматическая проверяющая система с архивом задач, на базе которой проводятся чемпионаты ACM по Дальневосточному региону, соревнования школьников, тренировки.

*Приватное поле* – поле с данными, доступными для просмотра только определенным группам пользователей.

* 1. Описание предметной области

Современное общество предоставляет много возможностей для повышения уровня образования, причем не только в учебных заведениях, но и дополнительными способами. Родители отдают своих детей на факультативы в школе, нанимают им репетиторов. Но одним из самых эффективных способов обучения является организация курсов на базе ВУЗов. Такие занятия ведут квалифицированные специалисты, чей уровень подготовки существенно отличается от школьных учителей. ВУЗы организовывают такие курсы для привлечения потенциальных студентов, так как у школьников, закончивших предлагаемые университетом курсы, появляется желание продолжить обучение именно в этом образовательном учреждении.

Ярким примером служит Дальневосточный Федеральный Университет, который славится своими одаренными студентами. В стенах ВУЗа существует множество курсов, повышающих уровень знаний и помогающих развивать свои способности.

Ежегодно университет выпускает специалистов в области информационных технологий. Но мало кто знает, что настоящие профессионалы своего дела появляются не сразу, а долгие годы совершенствуются на курсах программирования, организованными кафедрой информатики, математического и компьютерного моделирования.

Набор учащихся 7-11 классов производится по результатам летней школы, школьных олимпиад различного уровня, а также по рекомендациям школьных преподавателей и после собеседования. При поступлении на курсы происходит проверка уровня подготовки потенциального участника, после чего его определяют в одну из трех ступеней:

1. *Подготовительная.* Данная ступень является обязательной. В группу попадают неподготовленные участники, не имеющие никакой начальной базы по программированию. На протяжение всего курса ребята изучают простейшие команды на языках Basic Pascal и С++, проходят типы переменных, основные понятия программирования и прочее, необходимое для дальнейшей работы. Практические занятия в основном имеют форму решения задач олимпиадного типа в системе проверки заданий CATS. Полностью закончив данный курс, участники проходят тестирование, после чего руководитель принимает решение о переводе участников в следующую ступень.[[3]](#imcs)
2. *Начальная.* Данный курс включает изучение следующих разделов:

* язык (ветвления, циклы, работа с файлами, среды разработки)
* функции, рекурсия
* перебор
* сортировка, поиск, кучи (сортировки обменом, пузырьком, вставками, быстрая сортировка, сортировка слиянием, куча, сортировка с помощью кучи, бинарный поиск)
* геометрические алгоритмы (точки, прямые, отрезки, их взаимное расположение, многоугольники и связанные с ними задачи)
* комбинаторика (основные понятия, генерация комбинаторных объектов)
* структуры данных, динамическая память, записи (связные списки, деревья, их реализация в статически выделенной памяти)
* жадные алгоритмы
* динамическое программирование
* графы (основные понятия, представление графа в памяти, обход в глубину, обход в ширину, связные компоненты)

1. *Высшая.* На данной ступени обучаются уже состоявшиеся программисты, прошедшие предыдущие две ступени подготовки. Учебная программа состоит из сложных задач, составленных либо преподавательским составом кафедры, либо заимствованных из различных олимпиад и чемпионатов по программированию. Обучающиеся в этой ступени студенты участвуют в олимпиадах высшего уровня (всероссийские, международные).

Также для подготовки юных программистов организаторы проводят летнюю школу по математике и программированию. Возможность посещать школу есть у школьников, окончивших 7-10 класс и желающих получить углубленные знания в области программирования. Курс обучения предполагает знания в объёме школьной программы. Занятия по программированию проводятся преподавателями и студентами кафедры информатики. При обучении делается упор на алгоритмические аспекты программирования, в качестве базовых языков используются Паскаль и C++. Практические занятия в основном имеют форму решения задач олимпиадного типа в системе проверки заданий CATS. На основании результатов летней школы проводится отбор с среднюю и старшую группу мастер-класса.

Помимо мастер-классов по программированию и летней школы в Дальневосточном Федеральном университете существует малая академия юных программистов, предназначенная для школьников. Набор учащихся 7-11 классов производится по желанию учащихся, а также по рекомендациям школьных преподавателей после собеседования. В отличие от мастер-класса, при обучении делается упор на основные понятия программирования, не требующие значительной предварительной подготовки. В качестве базового языка используется Python. Практические занятия в основном имеют форму разработки интерактивных программ.

* 1. Существующие проблемы

Исходя из того, что единой системы контроля мастер-классов не существует, возникает ряд проблем:

1. *Отсутствие электронного списка участников, рейтинга и контроля посещаемости мастер-классов*

По результатам опроса преподавателей мастер-классов ДВФУ было выяснено, что на данный момент преподаватели не ведут учет посещаемости школьников, а текущий рейтинг участники просматривают в тестирующей системе CATS или с помощью редактора электронных таблиц Microsoft Excel.

Из-за отсутствия полных и списков участников преподаватели не имеют возможности заполнять рейтинг и вести контроль посещаемости. Также в списке должна быть информация для связи с ними.

* 1. *Проблема отчисления*

Из-за отсутствия рейтинга и учета посещаемости возникают трудности с выявлением участников, которые прогуливают занятия, либо же не набирают достаточное количество баллов для дальнейшего обучения. Поэтому школьники, пропустившие 3 и более занятия без уважительной причины или не имеющие определенного количества баллов в рейтинге, продолжают числиться в списках.

1. *Проблема оповещения и связи с участниками*

На данный момент система оповещения представляет собой группу в социальной сети «ВКонтакте», в которой преподаватели размещают записи о различных изменениях в расписании, таких как:

* начало занятий
* перенос занятий
* отмена занятий

Этот метод неудобен тем, что, во-первых, не все участники мастер-классов зарегистрированы в данной социальной сети. Во-вторых, далеко не всегда имеется возможность проверить изменения в группе (отсутствие интернета, к примеру). Еще одной причиной неудобства данного способа оповещения является то, что обновления в группе не сопровождаются уведомлениями, следовательно, информация может быть несвоевременно донесена до участников.

Также, ввиду отсутствия электронной базы данных, у преподавателей возникает проблема с формированием списка контактов участников и связи с обучаемыми в случае необходимости.

1. *Проблема «текучки»*

Мастер-классы дают возможность школьникам попробовать свои силы и определиться в правильности решения изучать программирование. Для первичного ознакомления с программой курса все потенциальные участники могут свободно посетить вводное занятие и сделать выбор. Часть из них уже после нескольких занятий понимают, что программирование не для них и забрасывают занятия, часто не сообщая о своем решении преподавателю. Таким образом возникает проблема учета состава в связи с «текучкой». В списках появляются «мертвые души», связаться с которыми довольно непросто из-за отсутствия контактов. Наиболее остро эта проблема стоит в подготовительной ступени, в меньшей степени – в начальной. В высшей ступени данной проблемы нет, так как все участники уже окончательно определились в своем желании посещать курсы.

1. *Проблема подбора компетентного преподавательского состава*

Специфика ведения мастер-классов требует определенных преподавательских качеств, а именно: способность донести информацию до обучающихся в понятном виде, всегда располагать актуальными сведениями, давать исчерпывающие ответы на вопросы и находить общий язык с каждым участником.

На данный момент все преподаватели мастер-классов являются студентами старших курсов ДВФУ, и, помимо руководительских обязанностей, имеют личную и учебную жизнь. Поэтому, некоторые из них не всегда способны справиться с нагрузками и грамотно распределить время. Но, если во время учебного семестра занятия проводятся в штатном режиме, то в период сессии (а иногда и несколько месяцев после) преподаватели пытаются закрыть свои собственные долги и у них просто не хватает времени на ведение мастер-классов. Из-за этого в первую очередь страдают участники, так как происходят срывы занятий и торможение учебного процесса. Также ни у кого из преподавателей нет педагогического образования и официального трудоустройства. Но так как они являются единственными кандидатами на данную должность, обе стороны (участники и преподаватели) продолжают испытывать некоторые неудобства.

* 1. Обзор существующих решений

Приведем сравнительную характеристику аналогичных и близких по тематике систем. (см. Таблица сравнительных характеристик)

Таблица 1. Таблица сравнительных характеристик

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Cats[[4]](#cats) | Blackboard[[5]](#blackboard) | Works | Webrate[[6]](#rating) | Сетевой Город. Образование [[7]](#nakhodka) | Дневник.ру [[8]](#dnevnikru) | Электронная школа Приморья [[9]](#dnevnikishkolapk) | Электронный дневник учащегося[[10]](#schoolconnect)  http://www.schoolconnect.ru/ | **Система контроля обучения** мастерклассов |
| Наличие рейтинга участников | + | - | + | + | + | + | + | + | + |
| Личный кабинет участника | + | + | - | - | + | + | + | + | + |
| Учет посещаемости участников | - | - | - | + | - | + | + | + | + |
| Просмотр в неавторизованном режиме | + | - | + | + | - | - | - | - | + |
| Просмотр рейтинга других участников | + | - | + | + | + | + | + | + | + |
| Просмотр домашних заданий/задач | + | + | + | - | + | + | + | - | + |

\*Сетевой Город. Образование – <http://city.nakhodka-edu.ru/> – комплексная информационная система, включает в себя электронные журналы и дневники, средства для планирования и мониторинга учебного процесса, оперативного общения между всеми его участниками.

\*Дневник.ру – <http://dnevnik.ru/> – единая образовательная сеть России, формирующая уникальную электронную среду для учителей, учеников и их родителей, сохранены все основные функции социальной сети: личные страницы, возможность добавлять друзей, отправка личных сообщений, группы и события и обмен файлами.

Есть возможность зарегистрироваться и протестировать систему в качестве гостя.

\*Электронная школа Приморья – <http://dnevniki.shkolapk.ru/> – наиболее полноценная система контроля образования, в том числе и дополнительного (включающее курсы программирования), дошкольного и коррекционного. Есть возможность бесплатного СМС-информирования.

С 1 января 2015 года, в связи с прекращением договорных отношений между ЗАО «Ланит ДВ» и департаментом образования и науки Приморского края, действие данной системы приостановлено

\*Электронный дневник учащегося (ЭДУ) – <http://www.schoolconnect.ru/> – единое информационное пространство, объединяющее школу, семьи учащихся и органы управления образованием. Имеет максимально удобный сервис для контроля качества образования.

**Вывод**:

Итак, все сайты имеют определенные недостатки в плане функциональности. Ни одна из представленных систем не подходят для выполнения поставленной цели, так как они либо ориентированы на обучение в школе, где совершенно другая система оценивания, либо у них отсутствует возможность подключить свою группу обучающихся.

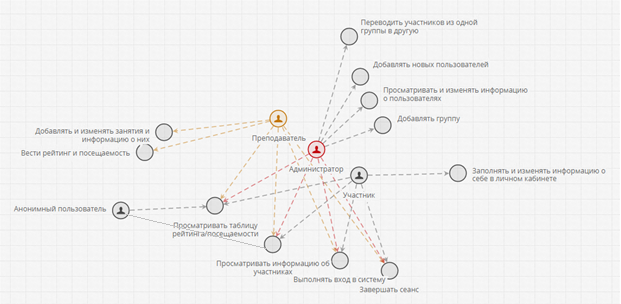
При реализации будут учтены все преимущества вышеизложенных систем, затем будет предпринят ряд мер по решению всех недостатков.

* 1. Неформальная постановка задачи

Цель работы – разработать и реализовать сайт, представляющий собой онлайн систему для контроля обучения на мастер-классах по программированию ДВФУ. Предполагается, что система автоматизирует работу преподавателей курсов и существенно упростит контроль обучения участников данных мастер-классов.

Требования к сайту:

* Посещаемость. Должна быть реализована сводная таблица, в которой будут учитываться посещения и пропуски занятий.
* Рейтинг. Система должна позволять вести рейтинг достижений участников для контроля и своевременного выявления проблем в обучении.
* Профиль. Для каждого участника должен быть реализован личный кабинет, в котором можно будет редактировать информацию о себе.



* Рисунок 1. Диаграмма вариантов использования

1. Требования к окружению
   1. Требования к аппаратному обеспечению

Для проведения занятий необходимо наличие компьютерного класса с сетевым оборудованием.

На ПК обязательно должен быть доступ к сети Интернет.

* 1. Требования к программному обеспечению

Сайт гарантированно работает на веб-браузере *Google Chrome v43.0.2.1* и выше

Тестирование на других браузерах не проводилось и работоспособность сайта не может быть гарантирована.

* 1. Требования к пользователям

В системе предусмотрены следующие роли пользователей:

* Администратор;
* Преподаватель;
* Участник;
* Анонимный пользователь;

Администратор отвечает за работу системы в целом. В обязанности администратора входит составление списка логин/пароль для каждого участника группы. Списки передаются преподавателю, а тот, в свою очередь, предоставляет данные для авторизации участникам. Каждый участник обязан войти в систему с выданным логином/паролем и, по желанию, заполнить информацию о себе в личном кабинете. При добавлении администратором участника в базу указывается ФИО участника, логин, пароль и группа, к которой он принадлежит. Также, по окончании учебного года или при досрочном переводе, администратор переводит каждого участника в следующую группу. Список участников, рекомендованных для перевода, предоставляется администратору преподавателем каждой группы лично. При переводе в другую группу участнику присваивается новая активная группа, а старая/законченная может быть переведена в статус «архивная». Но просмотр рейтинга и посещаемость неактивных (архивных) групп остается доступным для всех видов пользователей.

Такая же система авторизации действует и для преподавателей. Преподаватели, вошедшие в систему, по умолчанию, обладают правами, предоставляющими возможность работы с таблицей рейтинга и таблицей посещаемостью (см. Функциональные требования).

Роль преподавателя заключается в ведении рейтинга и контроля посещаемости участников. Преподаватель также имеет возможность создавать занятие и указывать информацию о нем. Так как преподаватели в группах часто заменяют друг друга, группа участников не привязывается к конкретному преподавателю, поэтому каждый преподаватель имеет возможность работать с таблицами рейтинга и посещаемости любой группы.

Анонимный пользователь (неавторизованный) имеет возможность просматривать профили пользователей, таблицу рейтинга и посещаемости каждой группы, а также ознакомляться с общедоступной информацией о мастер-классах, размещенной на главной странице. А просмотр заданий?

Участник, после авторизации, помимо перечисленных возможностей анонимного пользователя имеет возможность редактировать информацию в личном кабинете. (Просмотр осуществляется переходом в личный кабинет по гиперссылке, доступной в таблице рейтинга/посещаемости и в меню группы).

* 1. Функциональные требования

Система должна предоставлять следующие возможности:

* Администратору:
* Создавать группу с указанием параметров:
* Год обучения;
* Ступень (подготовительная, начальная, высшая);
* Преподаватель;
* Возможность просматривать и изменять всю информацию о пользователях;
* Возможность добавления новых пользователей (администраторов, преподавателей и участников) указывая:
* логин;
* пароль;
* ФИО;
* группа (для участников);
* Возможность переводить участников из группы в группу (присваивать статус «активная» новой группе и статус «архив» - старой);
* Возможность экспортирования списка участников определенной группы с указанием пары логин-пароль для каждого;
* Возможность просматривать список участников без группы;
* Возможность массового создания участников по списку из ф;
* Возможность менять преподавателя группы;
* Предоставлять преподавателю:
* Возможность просматривать приватную информацию об участниках;
* Возможность добавлять, редактировать и удалять занятие, а именно:
* Название занятия;
* Указывать дату проведения занятия;
* Добавлять задание, указывая:
* Описание задания;
* Максимальный балл за задание;
* Возможность работы с таблицей рейтинга:
* Выбирать группу учеников мастер-класса (год, ступень);
* Возможность проставлять\изменять баллы конкретному участнику за определенное занятие;
* Возможность работы с таблицей посещаемости:
* Выбирать группу учеников мастер-класса (год, ступень);
* Отмечать посещение участниками определенного занятия (+\-);
* Предоставить участнику:
* Возможность просматривать информацию о других участниках (за исключением приватных полей);
* Возможность работы с таблицей рейтинга:
* Выбирать группу;
* Просматривать баллы выбранной группы;
* Возможность просматривать информацию о конкретном занятии;
* Возможность работы с таблицей посещаемости:
* Выбирать группу;
* Просматривать посещаемость выбранной группы;
* Возможность заполнять и изменять информацию в личном кабинете, а именно:
* Телефон;
* Школа;
* Класс;
* Место жительства \ населенный пункт;
* Дата рождения;
* ФИО родителя;
* Телефон (приватное поле);
* Адрес электронной почты (приватное поле);
* Телефон родителя (приватное поле);
* Предоставить возможность для групп Преподаватель, Участник, Администратор:
  + Авторизовываться в системе, а именно указывать:
* Логин;
* Пароль;
  + Изменять пароль после входа в систему;
  + Завершать сеанс;

Возможности администратора включают в себя все возможности преподавателя и участника. Возможности преподавателя включают в себя все возможности участника.

1. Спецификация данных
   1. Диаграмма базы данных

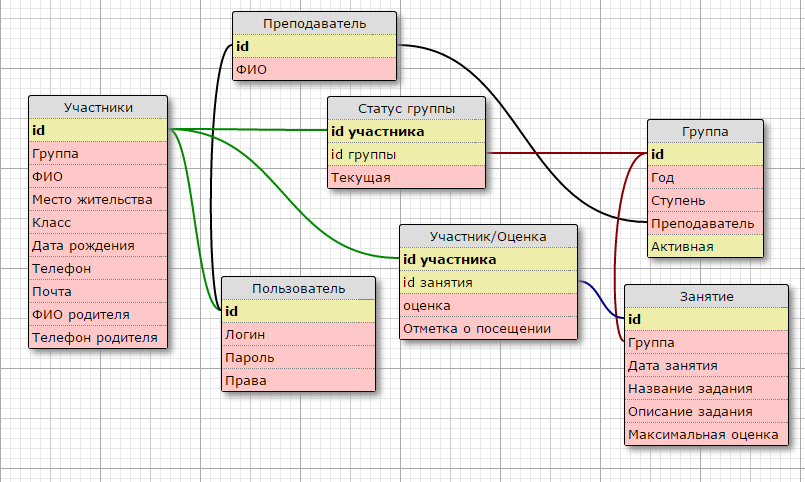


Рисунок 2. Диаграмма базы данных

* 1. Описание сущностей

Далее будут приведены сущности (таблицы), имеющие смысл в предметной области.

* + 1. Сущность «Пользователи»

Предназначена для хранения авторизационных данных.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название** | **Тип** | **Ограничения** | **Обязательность** | **Можно редактировать** | **Уникальность** | **По умолчанию** |
| Логин | Строка | 40 символов | Да | Да | Да | - |
| Пароль | Строка | 40 символов | Да | Да | Нет | - |
| Права | Целое число | 0..2 | Да | Нет | Не | - |

* + 1. Сущность «Участники»

Предназначена для хранения данных об участниках.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название** | **Тип** | **Ограничения** | **Обязательность** | **Можно редактировать** | **Уникальность** | **По умолчанию** |
| ФИО | Строка | 80 символов | Да | Да | Нет | - |
| Место жительства | Строка | 200 символов | Нет | Да | Нет | - |
| Школа | Строка | 200 символов | Нет | Да | Нет | - |
| Класс | Целое число | - | Нет | Да | Нет | - |
| Дата рождения | Дата | - | Нет | Да | Нет | - |
| Телефон | Строка | 15 символов | Нет | Да | Нет | - |
| E-mail | Строка | 40 символов | Нет | Да | Нет | - |
| ФИО родителя | Строка | 80 символов | Нет | Да | Нет | - |
| Телефон родителя | Строка | 15 символов | Нет | Да | Нет | - |

* + 1. Сущность «Преподаватели»

Предназначена для хранения данных о преподавателях.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название** | **Тип** | **Ограничения** | **Обязательность** | **Можно редактировать** | **Уникальность** | **По умолчанию** |
| ФИО | Строка | 80 символов | Да | Да | Нет | - |

* + 1. Сущность «Группы»

Предназначена для хранения информации об имеющихся группах обучения.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название** | **Тип** | **Ограничения** | **Обязательность** | **Можно редактировать** | **Уникальность** | **По умолчанию** |
| Год | Целое число | - | Да | Нет | Нет | - |
| Название ступени | Строка | 20 символов | Да | Нет | Нет | - |
| Преподаватель | Ссылка на сущность «Преподаватели» | - | Нет | Да | Нет | - |
| Активная | Логический | - | Да | Да | Нет | - |

* + 1. Сущность «Занятия»

Предназначена для хранения данных о занятиях мастер-классов.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название** | **Тип** | **Ограничения** | **Обязательность** | **Можно редактировать** | **Уникальность** | **По умолчанию** |
| Группа | Ссылка на сущность «Группы» |  | Да | Нет | Нет | - |
| Дата занятия | Дата |  | Нет | Да | Нет | - |
| Название занятия/задания | Строка | 50 символов | Нет | Да | Нет | - |
| Описание занятия/задания | Строка | 500 символов | Нет | Да | Нет | - |
| Максимальная оценка за задание | Целое число |  | Да | Да | Нет | 0 |

* + 1. Сущность «Статус группы»

Предназначена для хранения данных об активных и архивных группах участника.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название** | **Тип** | **Ограничения** | **Обязательность** | **Можно редактировать** | **Уникальность** | **По умолчанию** |
| Участник | Ссылка на сущность «Участники» |  | Да | Нет | Да | - |
| Группа | Ссылка на сущность «Группы» |  | Да | Нет | - |
| Текущая | Логический |  | Да | Да | - |

* + 1. Сущность «Участник/Оценка»

Предназначена для хранения данных об успеваемости и посещаемости участников.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Название** | **Тип** | **Ограничения** | **Обязательность** | **Можно редактировать** | **Уникальность** | **По умолчанию** |
| Участник | Ссылка на сущность «Участники» |  | Да | Нет | Да | - |
| Занятие | Ссылка на сущность «Занятия» |  | Да | Нет | - |
| Оценка | Целое число |  | Нет | Да | Нет | - |
| Отметка о посещении | Логический |  | Нет | Да | Нет | - |

1. Проект
   1. Средства реализации

Для реализации был выбран язык программирования PHP 5.3, JavaScript с использованием JSON, HTML, СУБД MySQL.

* 1. Структуры данных
     1. Таблица «users»

|  |  |
| --- | --- |
| **Имя поля** | **Тип данных** |
| id | int |
| login | varchar(80) |
| pass | varchar(80) |
| rights | int |

* + 1. Таблица «students»

|  |  |
| --- | --- |
| **Имя поля** | **Тип** |
| id | int |
| fio | varchar(80) |
| placeofliving | varchar(200) |
| school | varchar(200) |
| grade | int |
| birthdate | date |
| phone | varchar(15) |
| email | varchar(40) |
| parentfio | varchar(80) |
| parentphone | varchar(15) |

* + 1. Таблица «teachers»

|  |  |
| --- | --- |
| **Имя поля** | **Тип данных** |
| id | int |
| fio | varchar(80) |

* + 1. Таблица «groups»

|  |  |
| --- | --- |
| **Имя поля** | **Тип** |
| id | int |
| year | int |
| grade | varcher(20) |
| teacher\_id | int |
| active | tinyint(1) |

* + 1. Таблица «tasks»

|  |  |
| --- | --- |
| **Имя поля** | **Тип** |
| id | int |
| group\_id | int |
| taskdate | date |
| taskname | varchar(50) |
| description | varchar(500) |
| maxrate | int |

* + 1. Таблица «groups\_students»

Предназначена для хранения данных об активных и архивных группах участника.

|  |  |
| --- | --- |
| **Имя поля** | **Тип** |
| id | int |
| student\_id | int |
| group\_id | int |
| active | tinyint(1) |

* + 1. Таблица «students\_rating»

Предназначена для хранения данных об успеваемости и посещаемости участников.

|  |  |
| --- | --- |
| **Имя поля** | **Тип** |
| student\_id | int |
| task\_id | int |
| rating | int |
| attendance | tinyint(1) |

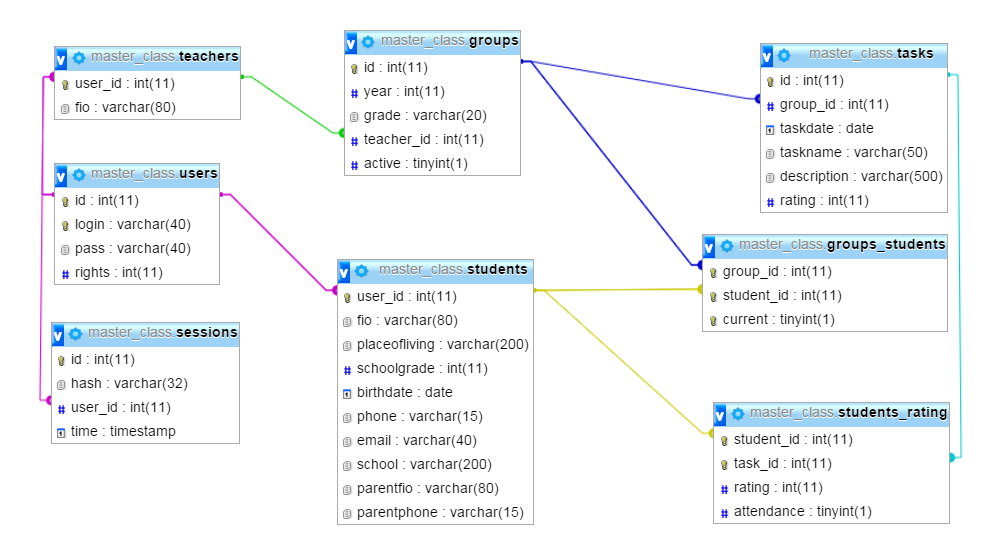


Рисунок 3. Полная диаграмма базы данных

* 1. Требование к интерфейсу

Система должна быть реализована в виде web-сайта. Должна включать в себя следующие страницы:

* Личный кабинет
* Настройки профиля
* Рейтинг
* Посещаемость
* Основной информационный раздел
* Панели администраторов для преподавателя и администратора
  1. Диаграмма состояний

Переходы в разделы осуществляются посредством нажатия на соответствующие пункты меню, доступные с любой страницы сайта.

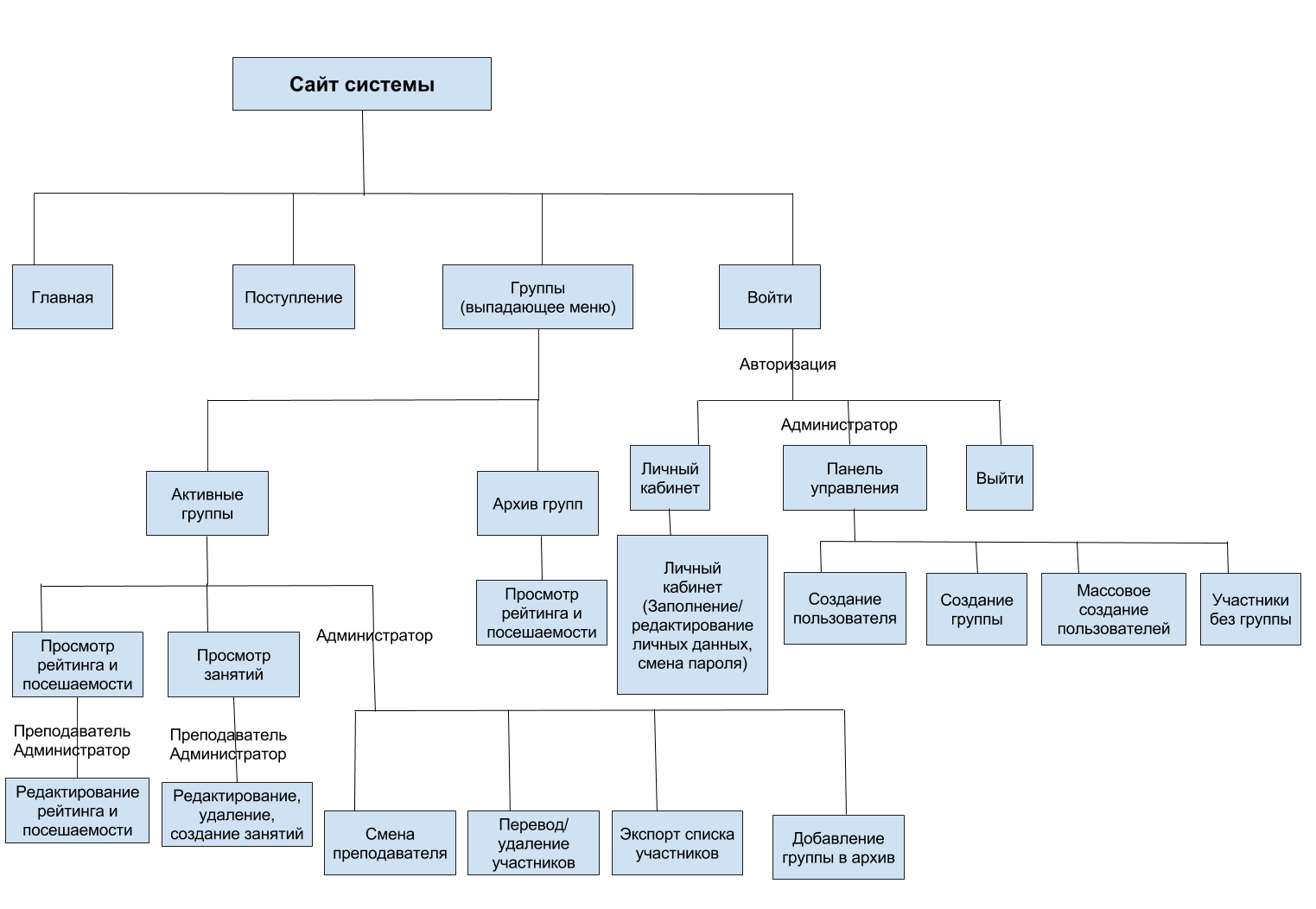


Рисунок 3. Диаграмма состояний

* 1. Проект интерфейса

Основной каркас можно разделить на следующие части: «шапка», горизонтальное меню, раздел авторизации, центральный блок и «подвал».

В «шапке» изначально присутствует название сайта, наименование кафедры, логотип университета и горизонтальная панель.

На горизонтальной панели располагается главное меню сайта (главная страница, ссылка на раздел рейтинга и посещаемости) и раздел авторизации.

В центральный блок выводится присущая каждой конкретно взятой странице информация. На начальной странице в данном блоке отображается раздел с информацией о мастер-классах.

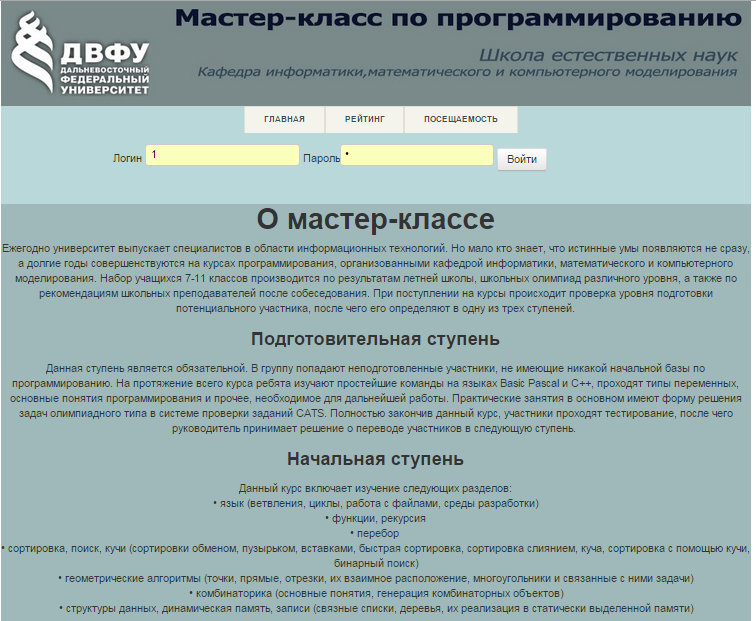


Рисунок 4. Главная страница

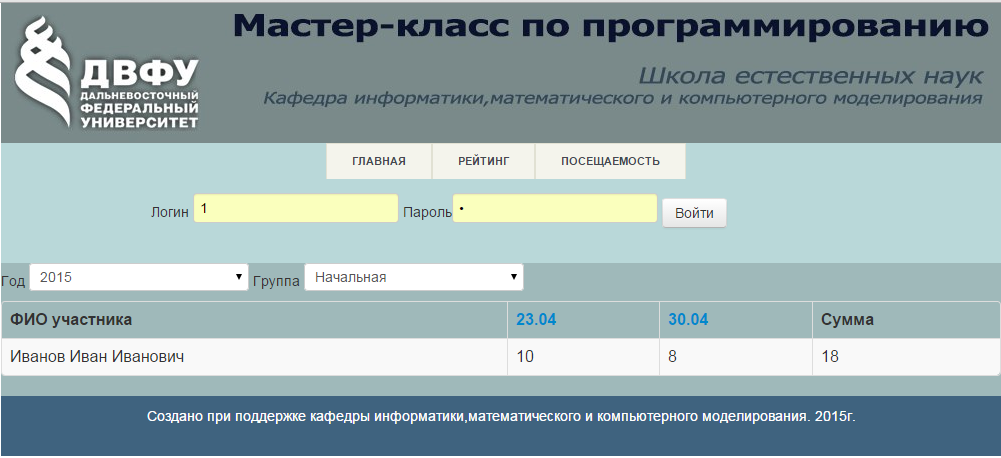


Рисунок 5. Рейтинг

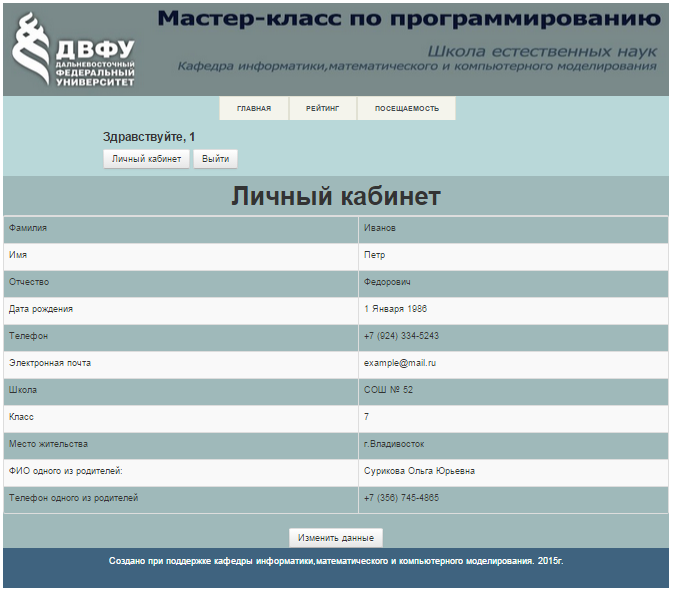


Рисунок 6. Личный кабинет

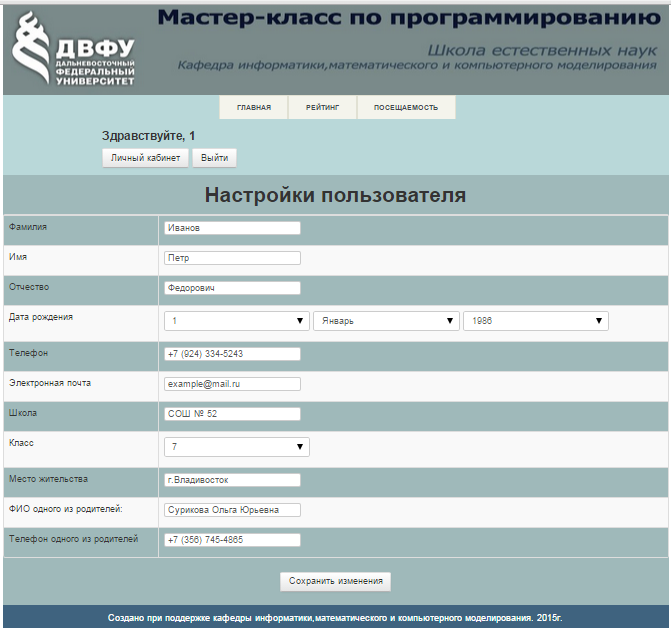


Рисунок 7. Изменение данных

Начальный интерфейс сайта может незначительно меняться согласно решениям разработчиков (цветовая схема, формы кнопок, стили, заголовки). Но структура системы остается неизменной.

В ходе реализации системы был переработан интерфейс сайта. Далее представлены скриншоты, иллюстрирующие внесенные изменения.

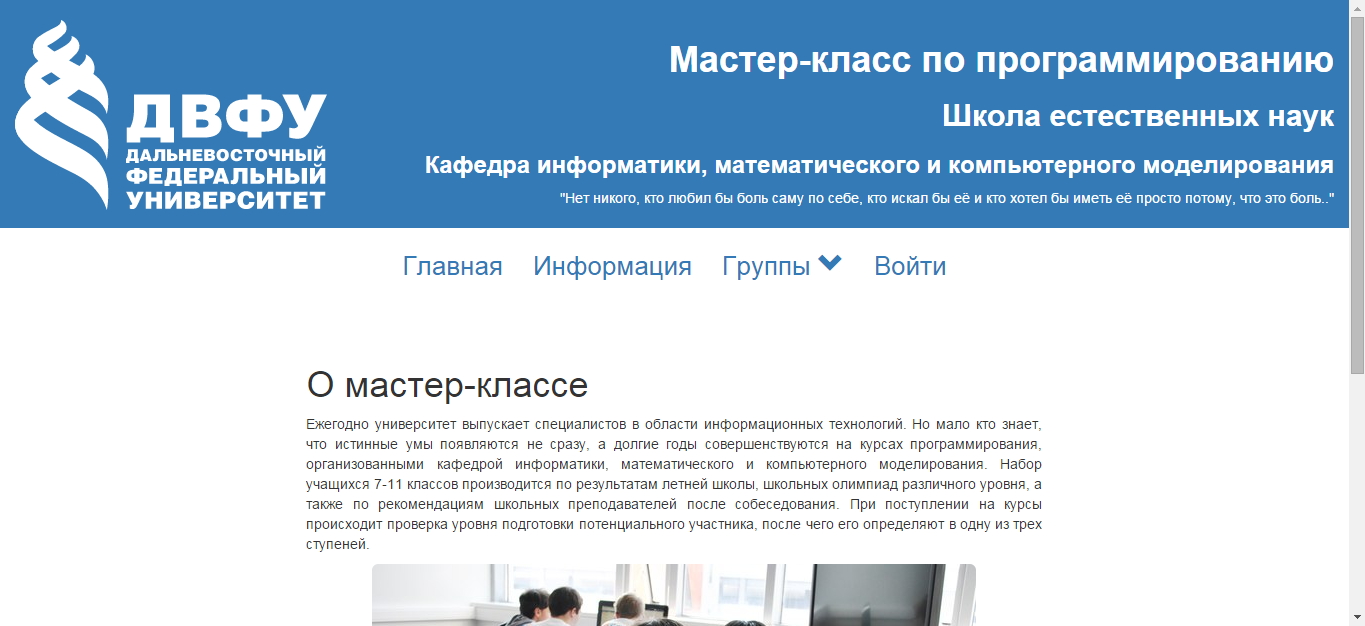


Рисунок . Главная страница

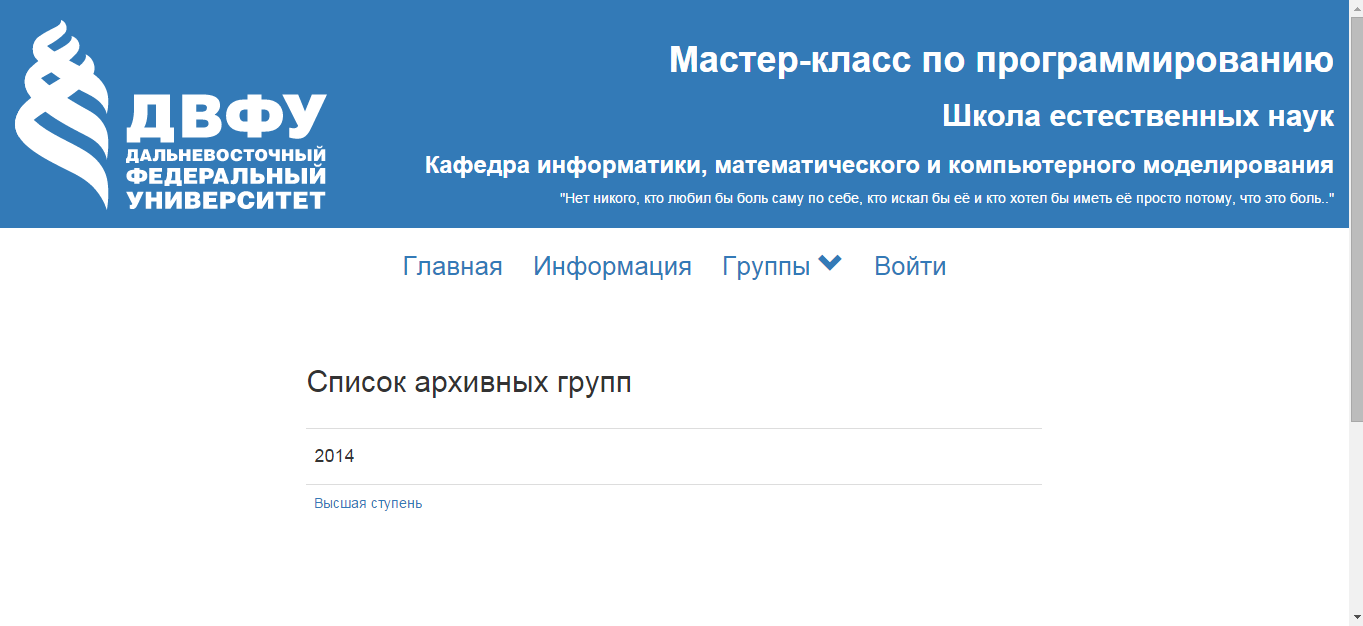


Рисунок .Список архивных групп

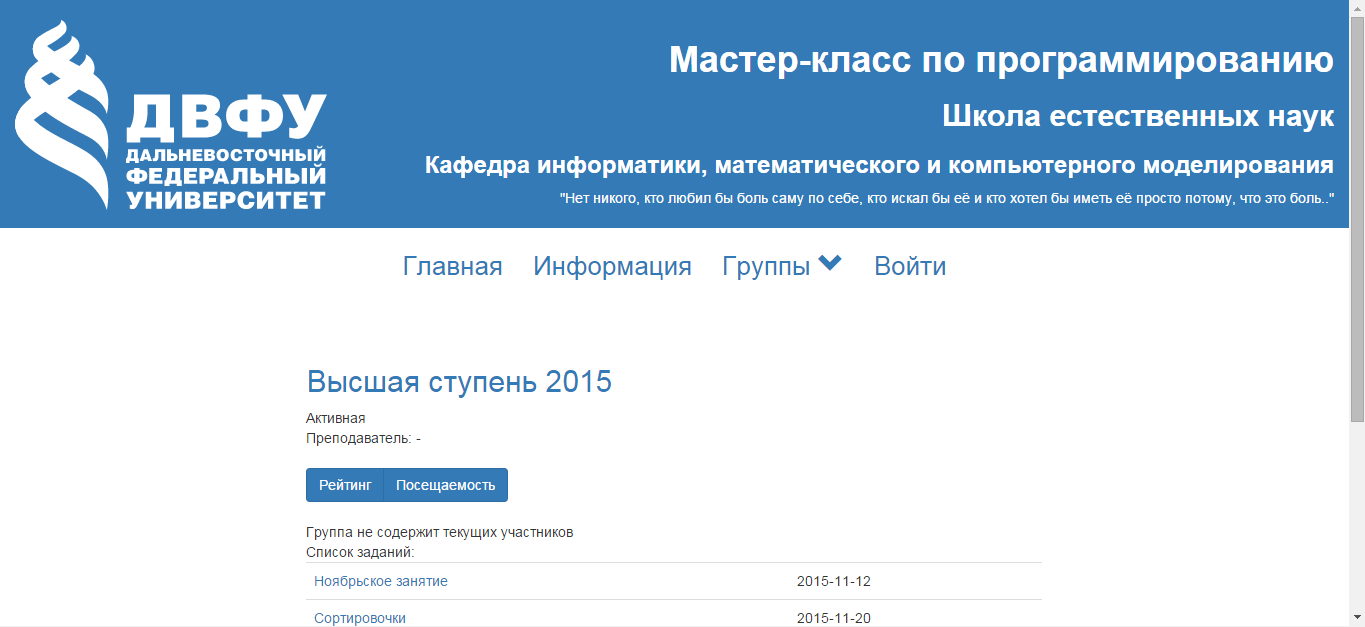


Рисунок .Страница группы

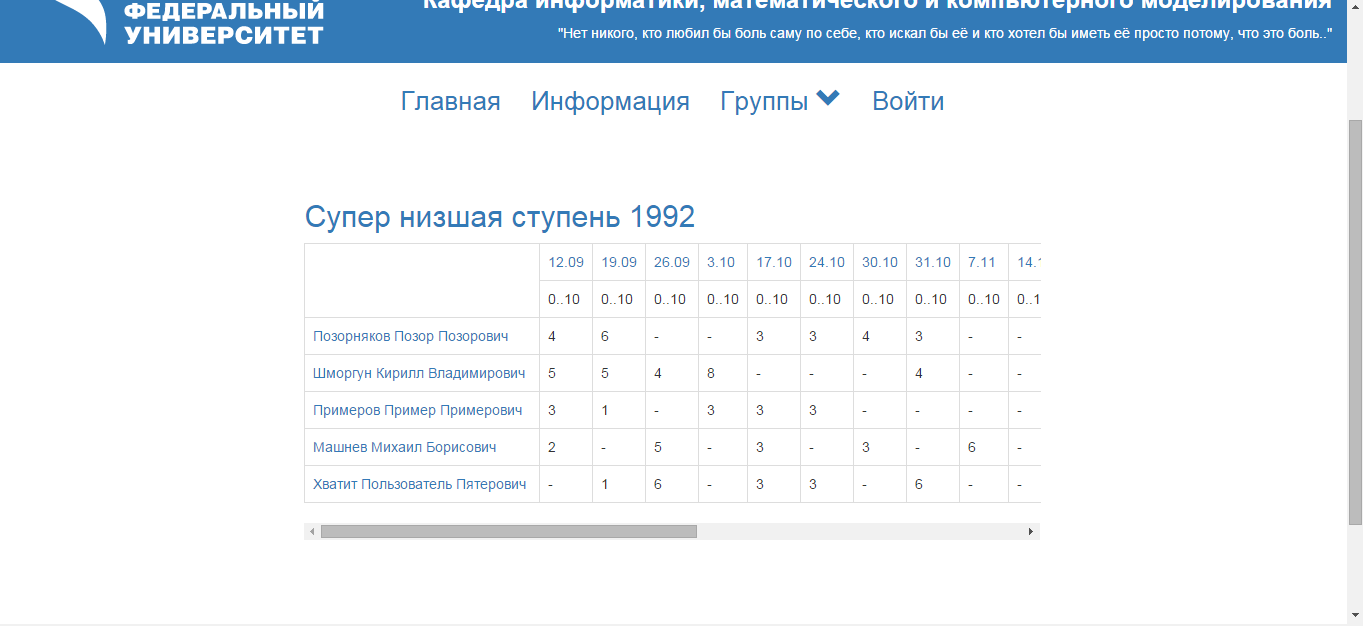


Рисунок .Рейтинг группы



Рисунок .Посещаемость группы

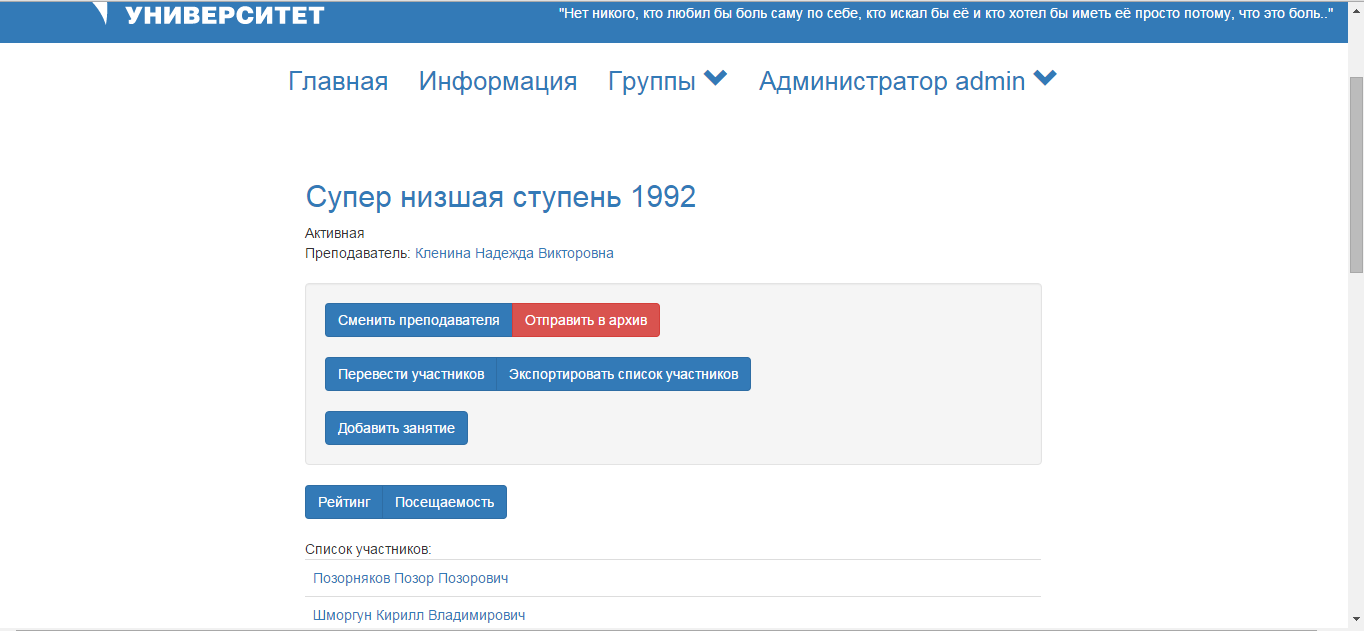


Рисунок .Меню группы (отображение для администратора)

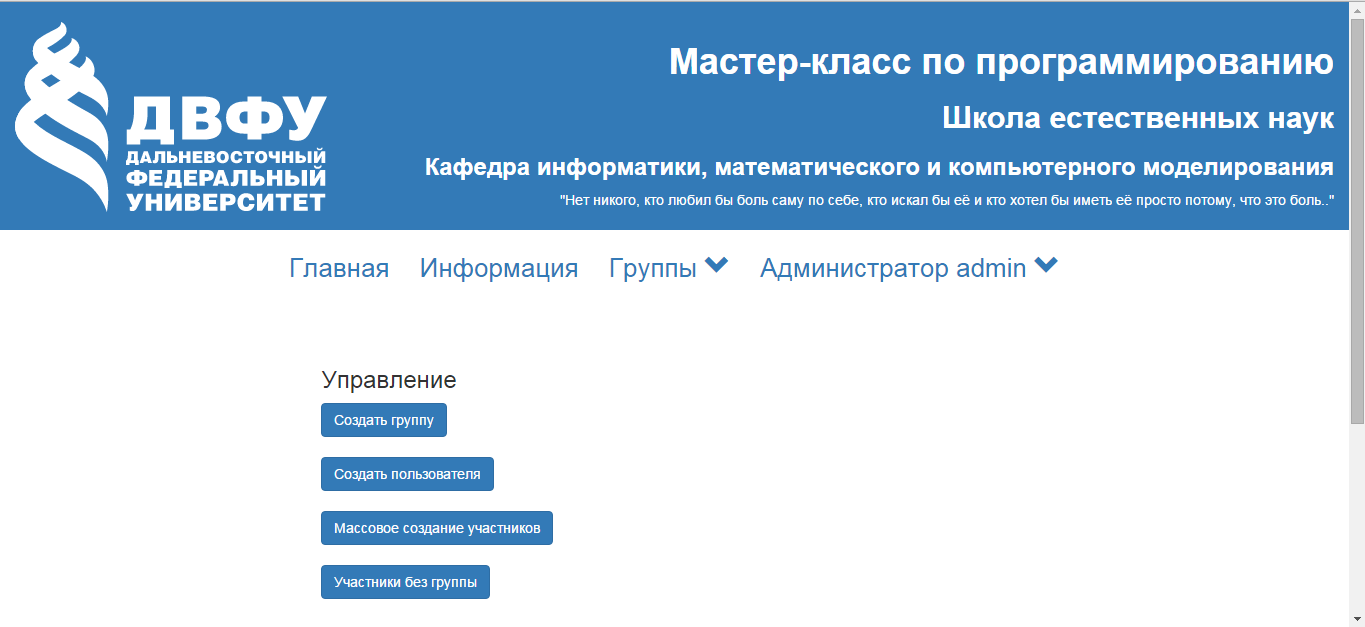


Рисунок . Панель управления (доступна администратору)

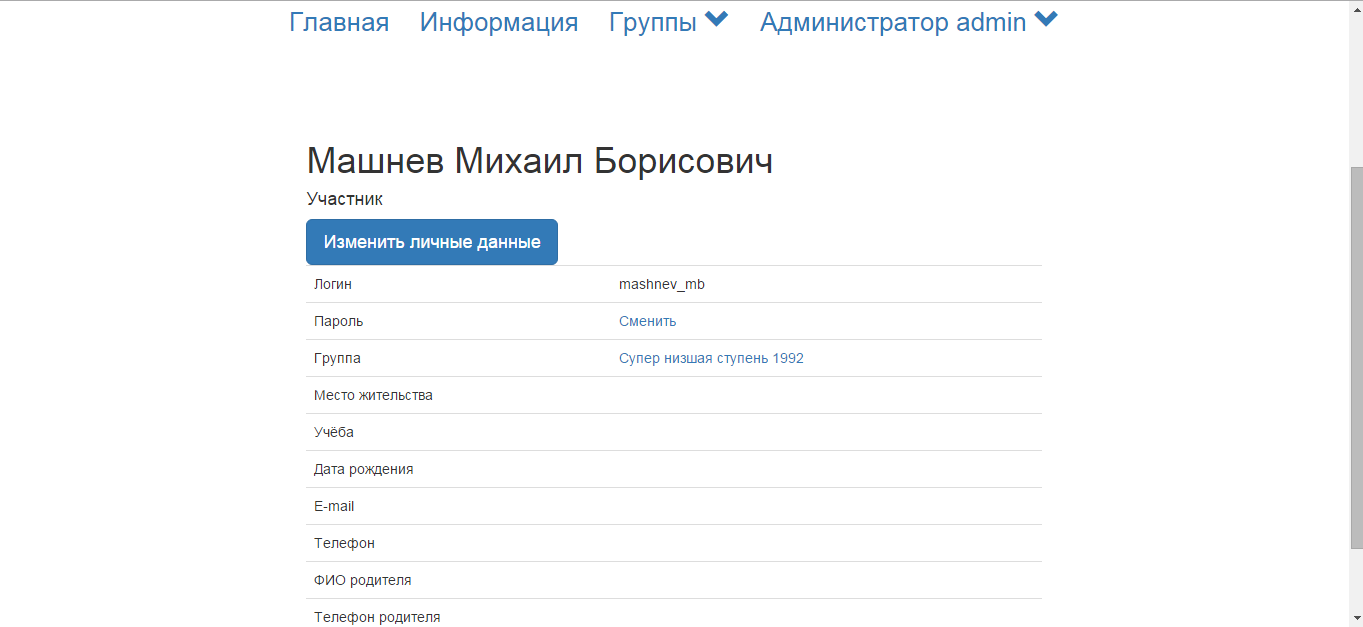


Рисунок .Личный кабинет участника (отображение для администратора)

1. Внесенные изменения

Изменено горизонтальное меню сайта. Теперь содержит кнопки «Главная», «Поступление», «Группы» (выпадающее меню со списком активных групп и архивом), «Войти» (для неавторизированных пользователей), «Участник/Преподаватель/Администратор [логин пользователя]» (для авторизированных пользователей).

Добавлена возможность массового создания участников для администратора по нажатию кнопки «Массовое создание участников», расположенной в админ-панели.

Добавлена возможность удаления участников из группы. Список участников без группы доступен администратору в админ-панели.

В меню группы участники, переведенные в другую группу или отчисленные из текущей, отображаются серым цветом.

Посещаемость в таблице посещаемости определяется по цвету ячейки (зеленый цвет для отображения посещений, красный цвет для отображения пропусков).

Заключение

Таким образом, в ходе данного исследования был изучен механизм работы мастер-классов и основные аспекты рейтингового учёта успеваемости. Были выявлены существующие проблемы и разработаны требования к системе.

Описанная в данном отчете система полностью реализована в виде web-сайта (30 уникальных web-страниц). Объем написанного кода: на JavaScript – 270 строк, HTML – 773, PHP – 3039.

Список литературы

1. <http://to-name.ru/biography/arhimed.htm>

1. <https://ru.wikipedia.org>

1. [http://imcs.dvfu.ru](http://imcs.dvfu.ru/works/kif.html)

1. [http://imcs.dvfu.ru/cats](http://imcs.dvfu.ru/cats/)

1. <https://bb.dvfu.ru>

1. <http://imcs.dvfu.ru/rating/>

1. <http://city.nakhodka-edu.ru/>

1. <http://dnevnik.ru/>

1. <http://dnevniki.shkolapk.ru/>

1. <http://www.schoolconnect.ru/>
2. <http://www.calculator888.ru/blog/biografiya/turing-alan.html>
3. [http://www.cyberforum.ru](http://www.cyberforum.ru/php-database/thread449046.html)
4. <http://htmlbook.ru>
5. <http://www.php.su>
6. <http://php.net>
7. <http://php-myadmin.ru/learning>