

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
Кафедра «Прикладная математика и информатика»

02.03.03 МАТЕМАТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ И АДМИНИСТРИРОВАНИЕ
ИНФОРМАЦИОННЫХ СИСТЕМ

ТЕХНОЛОГИЯ ПРОГРАММИРОВАНИЯ

БАКАЛАВРСКАЯ РАБОТА

на тему «Разработка веб-тренажера по дисциплине «Теория вероятностей
и математическая статистика»»»

Студент(ка) _____ Аминова А.И. _____

Руководитель _____ Туищев А.И. _____

Допустить к защите

Заведующий кафедрой к.тех.н, доцент, А.В. Очеповский _____

« _____ » _____ 20 _____ г.

Тольятти 2016

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
Кафедра «Прикладная математика и информатика»

УТВЕРЖДАЮ
Зав. кафедрой «Прикладная
математика и информатика»
_____ А.В. Очеповский

«_____» _____ 2016 г.

ЗАДАНИЕ
на выполнение бакалаврской работы

Студент (ка) _____ Аминова Анна Игоревна _____

1. Тема: Разработка веб-тренажера по дисциплине "Теория вероятностей и математическая статистика"
2. Срок сдачи студентом законченной выпускной квалификационной работы 19.06.2016
3. Требования к архитектуре: трехзвенная «клиент-серверная», основанная на использовании веб-технологий; сетевой режим работы (интернет); режим функционирования: круглосуточный (365/24/7); требования к технической реализации: в качестве СУБД нужно использовать MySQL, в качестве средства реализации использовать языки программирования HTML и PHP.
4. Содержание выпускной квалификационной работы (перечень подлежащих разработке вопросов, разделов): проанализировать научную и учебную литературу по проблеме реализации обучающего тренажера; проанализировать основные принципы создания веб-тренажеров; проанализировать аналогичные системы для определения технологии разработки программного приложения; описать математическую модель разрабатываемого программного приложения; создать структурную и функциональную модели веб-тренажера; обосновать выбор средств реализации приложения; описать алгоритмы работы веб-тренажера; реализовать приложение выбранными программными средствами; протестировать программное приложение; апробировать реализованный программный продукт.
5. Ориентировочный перечень графического и иллюстративного материала блок-схемы выполнения основных процессов программного приложения; графики и диаграммы, поясняющие результат работы приложения; экранные формы, демонстрирующие основные возможности работы приложения; презентация.

6. Дата выдачи задания « 11 » января 2016 г.

Заказчик, старший
преподаватель кафедры
«ПМИ»

М.А. Тренина

Руководитель выпускной
квалификационной работы

А.И. Туищев

Задание принял к исполнению

А.И. Аминова

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
Кафедра «Прикладная математика и информатика»

УТВЕРЖДАЮ
Зав. кафедрой «Прикладная
математика и информатика»
_____ А.В. Очеповский

«_____» _____ 2016 г.

КАЛЕНДАРНЫЙ ПЛАН
выполнения бакалаврской работы

Студента(ки) _____ Аминовой Анны Игоревны
по теме Разработка веб-тренажера по дисциплине "Теория вероятностей и
математическая статистика"

Наименование раздела работы	Плановый срок выполнения раздела	Фактический срок выполнения раздела	Отметка о выполнении	Подпись руководителя
Выбор и утверждение темы ВКР	13.01.2016	13.01.2016	Выполнено	
Анализ научной и учебной литературы по проблеме реализации обучающего тренажера	25.01.2016	25.01.2016	Выполнено	
Анализ основных принципов создания веб-тренажеров;	1.02.2016	1.02.2016	Выполнено	
Описание технологии создания веб-тренажера	8.02.2016	8.02.2016	Выполнено	
Анализ аналогичных систем для определения технологии разработки программного приложения	15.02.2016	15.02.2016	Выполнено	
Описание математической модели разрабатываемого программного приложения	27.02.2016	27.02.2016	Выполнено	
Создание структурной и функциональной модели	20.03.2016	20.03.2016	Выполнено	

веб-тренажера				
Обоснование выбора средств реализации приложения	22.03.2016	22.03.2016	Выполнено	
Описание алгоритмов работы веб-тренажера	5.04.2016	5.04.2016	Выполнено	
Реализация приложения выбранными программными средствами	28.04.2016	28.04.2016	Выполнено	
Описание контрольного примера работы созданного программного приложения	5.05.2016	5.05.2016	Выполнено	
Тестирование модуля программного приложения	10.05.2016	10.05.2016	Выполнено	
Апробация реализованного программного продукта.	18.05.2016	18.05.2016	Выполнено	
Предзащита ВКР	25.05.2016	25.05.2016	Выполнено	
Исправление замечаний по ВКР	29.05.2016	29.05.2016	Выполнено	
Проверка ВКР в системе «Антиплагиат.ВУЗ»	12.06.2016	12.06.2016	Выполнено	
Подготовка документов к защите	15.06.2016	15.06.2016	Выполнено	
Сдача пояснительной записки ВКР и реализованного программного приложения	19.06.2016	19.06.2016	Выполнено	

Руководитель выпускной
квалификационной работы

А.И. Туищев

Задание принял к исполнению

А.И. Аминова

Аннотация

Актуальность темы данной работы обусловлена информационным взрывом, который породил большое количество проблем, главнейшей из которых считается проблема организации процесса обучения. Определенный интерес представляют вопросы, связанные с подготовкой аудио и видео визуальных средств обучения, способных выступать основным помощником учителю.

Бакалаврская работа состоит из 52 страниц В работе: рисунков - 26, список литературы состоит из 34 источников.

Бакалаврская работа посвящена созданию и реализации веб-тренажера по дисциплине «Теория вероятностей и математическая статистика». Она состоит из введения, трех глав и заключения.

Во введении обосновывается актуальность темы исследования, формулируются цели и задачи, которые необходимо решить для достижения поставленной цели.

В первой главе рассматривается понятие и сущность веб-тренажера, его структура и основное содержание. Также производится анализ существующих веб-тренажеров и выявляются требования к разрабатываемому тренажеру.

Во второй главе осуществляется моделирование веб-тренажера (моделирование архитектуры, функциональное моделирование). Также рассматривается концептуальная и математическая модели веб-тренажера.

В третьей главе показывается реализация веб-тренажера по дисциплине «Теория вероятностей и математическая статистика». Рассматривается Веб-тренажер как средство обучения дисциплине, производится выбор средств реализации интерактивного веб-тренажера, также описание основного принципа работы реализуемого веб-тренажера

В заключении сделаны основные выводы и итоги по проделанной работе.

Оглавление

Введение.....	3
Глава 1 Теоретические основы построения веб-тренажера.....	5
1.1 Понятие веб-тренажера как обучающей электронной системы: сущность и этапы его построения	5
1.2. Структура и основное содержание веб-тренажера	7
1.3 Анализ существующих веб-тренажеров.....	9
1.4 Требования к разрабатываемому тренажеру	12
Глава 2 Моделирование веб-тренажера по дисциплине «Теория вероятностей и математическая статистика»	17
2.1 Моделирование архитектуры обучающей системы «Веб-тренажер».....	17
2.2 Функциональное моделирование обучающей системы «Веб-тренажер»..	20
2.3 Концептуальная модель веб-тренажера	22
2.4 Моделирование базы данных веб-тренажера	24
2.5 Математическая модель обучающей системы «Веб-тренажер».....	26
Глава 3 Реализация веб-тренажера по дисциплине «Теория вероятностей и математическая статистика»	32
3.1 Функциональное моделирование веб-тренажера как средства обучения дисциплине	32
3.2 Выбор средств реализации интерактивного веб-тренажера	35
3.3 Описание основного принципа работы реализуемого веб-тренажера.....	38
Заключение	48
Список используемой литературы	49

Введение

В образовательных учреждениях огромное внимание уделяется компьютерному сопровождению учебных занятий, для которых сегодня представлен огромный выбор тестирующих и обучающих программ по различным дисциплинам образовательного процесса.

Проведенные исследования использования тестирующих и обучающих программ по различным учебным курсам демонстрируют, что их использование способствует повышению интереса по направлению подготовки и изучаемому предмету, а также успеваемости по нему. Эти интерактивные обучающие средства предоставляют возможность любому студенту обучаться по индивидуальному плану, осуществлять самоконтроль и тем самым выстраивать индивидуальную образовательную траекторию. Этому способствует и то, что при использовании обучающих программ компьютер временно является исполнителем функций преподавателя

В сфере образования **актуальным** становится разработка компьютерных программ – электронных веб-тренажеров по различным дисциплинам.

Цель бакалаврской работы - разработка веб-тренажера по дисциплине «Теория вероятностей и математическая статистика».

Объект исследования – технология создания тренажера по дисциплине.

Предмет исследования - построение автоматизированного веб-тренажера.

Для реализации поставленной цели необходимо решить следующие **задачи**:

1. Проанализировать учебную, научную и методическую литературу по проблеме создания автоматизированного веб-тренажеров.

2. Сформулировать основные требования к созданию веб-тренажера и разработать его структуру на основе выделенных требований.

3. Проанализировать существующие для выделения основных компонентов, которые будут положены в основу веб-тренажера по дисциплине «Теория вероятностей и математическая статистика».

4. Обосновать выбор программного средства реализации веб-тренажера по дисциплине «Теория вероятностей и математическая статистика».

5. Разработать веб-тренажер по дисциплине «Теория вероятностей и математическая статистика» с соответствующим теоретическим и практическим материалом.

6. Обосновать необходимость применения веб-тренажера при изучении дисциплины «Теория вероятностей и математическая статистика».

Бакалаврская работа состоит из введения, двух глав, и заключения.

Во введении обосновывается актуальность выбранной темы, формируется проблема, выделяются цели и задачи.

В первой главе описаны теоретические основы построения веб-тренажера, включающие определение основной структуры и содержания. Приведен анализ существующих веб-тренажеров для выявления основных требований к разрабатываемому программному продукту.

Во второй главе представлены этапы проектирования и реализации веб-тренажера, которые затрагивают: моделирование структуры и реализацию на основе представления теоретического материала, практикума и системы контроля знаний.

В третьей главе дано обоснование выбора средств реализации веб-тренажера, приведено общее описание работы с ним.

В заключении сделаны основные выводы и итоги по проделанной работе.

Глава 1 Теоретические основы построения веб-тренажера

1.1 Понятие веб-тренажера как обучающей электронной системы: сущность и этапы его построения

В настоящее время для более полного изучения дисциплин, обучающимися используются веб-тренажеры. Существенное уменьшение финансовых затрат на обучение в комплексе с широкими возможностями веб-тренажера делают это направление привлекательным для любых пользователей, в том числе для университетов. Они в свою очередь пытаются обеспечить доступ любых пользователей к обучающей информации через глобальную сеть Интернет, тем самым решая проблему обучения студентов.

Из различных источников следуют разные определения электронного интерактивного обучающего средства - веб-тренажера [8, 10, 15, 23, 26]:

- учебное издание, содержащее материал, систематизированный по соответствующей научно-практической области знаний. Оно предназначено для овладения студентами знаниями, умениями и навыками в этой области;
- совокупность мультимедийной информации, а также печатной документации пользователя;
- совокупность мультимедийной информации, представленной в электронной форме и снабженной разветвленной системой связей, которые разрешают переходить от фрагмента к фрагменту в соответствии с иерархией фрагментов.

Исходя из представленных определений, можно сделать вывод что веб-тренажер - это учебное приложение содержащее набор тематического теоретического материала в комплексе с тестированием по предоставленным темам.

Компьютерный обучающий тренажер является обучающей системой, которая в свою очередь разделена на: подсистему управления, подсистему обучения и подсистему контроля. Они предоставляют материал для обучения, а также с помощью них можно самостоятельно изучить и отработать все знания и навыки по данной дисциплине. Также благодаря им обеспечивается управление

преподавателя-инструктора, а студентам предоставляется возможность самостоятельного управления обучением.

Основные задачи, которые решаются с помощью компьютерных тренажеров – теоретическая подготовка обучаемых, отработка ими практических навыков и знаний, а также их проверка и тестирование.

Компьютерный обучающий тренажер может быть построен на основании таких принципов как: модульность, самостоятельной организации образования, адаптивности и реализации [25].

Модульность является завершенным продуктом каждой процедуры, реализованной в тренажере и может использоваться отдельно.

Самостоятельная организация образования является взаимосвязанной последовательностью каждого элемента, т.е. разобрав один термин, проблему, вопрос студент переходит к следующему.

Адаптивность является возможностью использования модулей и методик, реализованных в веб-ориентированном интерактивном тренажере, для упорядочивания знаний.

Реализуемость является возможностью применения как для ручного, так и для машинного варианта в комплекте с простыми программными средствами.

Веб-тренажер представляет собой инновационный обучающий ресурс в области образовательных технологий и обеспечивает формирование практических навыков применения полученных знаний и умений на основе приближения обучения к реальным процессам.

Проанализируем ключевые стадии разработки веб-тренажера [28]:

1. Нахождение источников. Во время разработки веб-тренажера лучше всего выбирать источниками те электронные и печатные издания, что больше подходят к стандартной программе. Они должны быть удобными для написания текстов и содержать как можно больше задач и примеров. Также важно чтоб они предоставлялись в удобных форматах (принцип собираемости).

2. Формирование оглавления и списка понятий. Производится анализ материала и распределение по разделам, состоящим из модулей, небольших по

объему, но замкнутых по смыслу, а также формируется список понятий, которых будет достаточно для понятия предмета.

3. Распределение информации по модулям и разделам. Распределяются тексты источников согласно оглавлению и структуре модулей; тексты, не включенные в списки, убираются, а те, которых нет в источниках - добавляются; находятся связи между модулями, а также другие гипертекстные связи. Таким образом, подготавливается проект гипертекста для компьютерной реализации.

4. Представление в электронном виде. При создании веб-тренажера текст появляется в электронном виде. В итоге формируется стандартный сайт с обучающими материалами, которым можно пользоваться в учебных целях.

5. Организация компьютерной поддержки. Вырабатываются вспомогательные инструкции по работе с интеллектуальным ядром веб-тренажера, предназначенные пользователям. Теперь веб-тренажер готов к дальнейшим улучшениям (визуализации и озвучиванию) благодаря мультимедийным средствам.

6. Подборка материалов для мультимедийной реализации. Совершенствуются методы объяснения некоторых утверждений и понятий и производится сортировка текстов.

Таким образом, внедрение веб-тренажеров в процесс обучения значительно упрощает самостоятельное изучение материалов по каким-либо дисциплинам, открывая тем самым новые возможности в процессе обучения.

1.2. Структура и основное содержание веб-тренажера

Структура веб-тренажера обуславливается тем, что электронные интерактивные тренажеры предназначены для организации самообучения и должны четко распределяться на разделы в определенной индивидуальной последовательности и иметь связи между собой [25, 28]. Должна быть взята во внимание последовательность изучаемого материала: теоретическая и

практическая части, итоговые задания, а также материалы для дополнительного образования.

Каждый веб-тренажер включает в себя следующие составляющие: [25]

- средства изучения теоретического материала по теме;
- средства поддержки практикумов и лабораторных работ;
- средства контроля знаний в виде тестовых систем;
- методические рекомендации по изучению темы;
- средства управления процессом изучения темы.

На рисунке 1.1 представлена структура веб-тренажера.

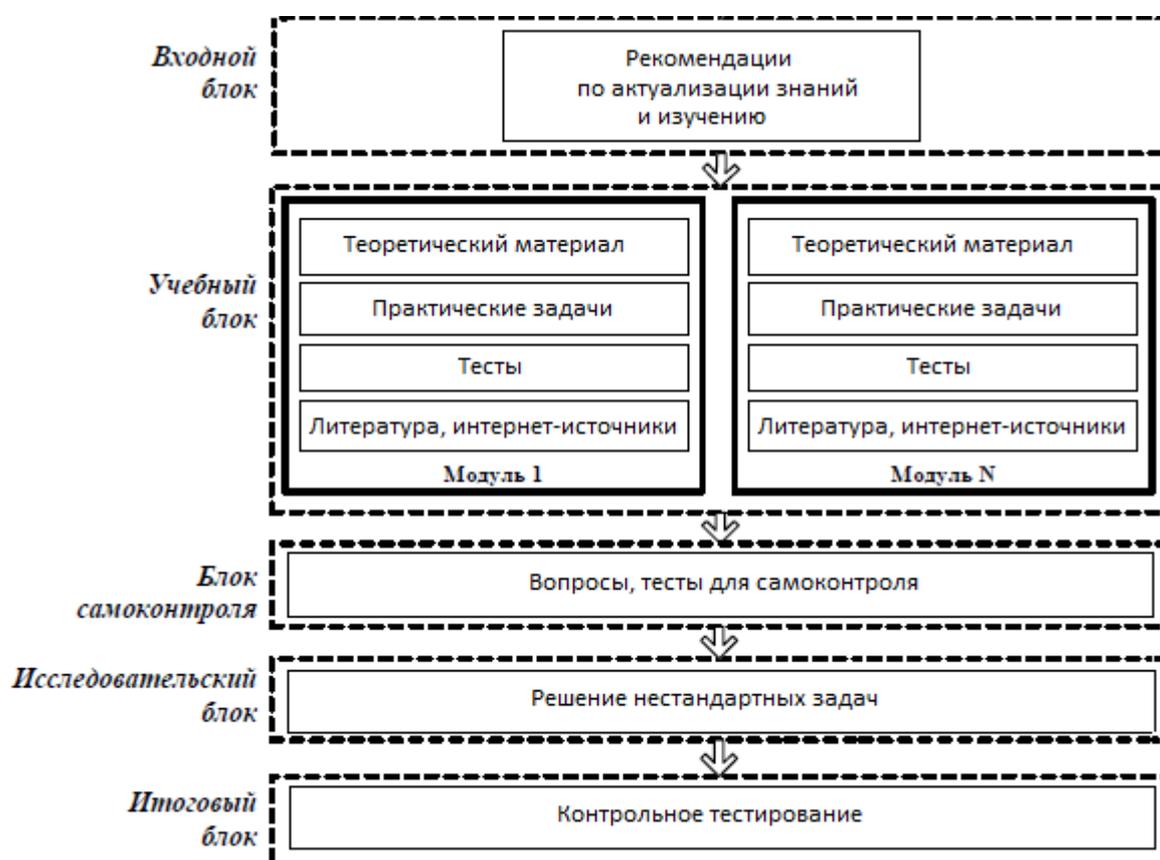


Рисунок 1.1 – Структура веб-тренажера

Представленная структура веб-тренажера состоит из следующих блоков, каждый из которых выполняет определенные задачи:

- входной блок: подготовка к изучению дисциплины;
- обучающий блок: обеспечение условий для изучения дисциплины;
- блок самоконтроля: развитие самостоятельности на основе средств самоконтроля и само корректировки;

- итоговый блок: оценка результатов обучения.

В интерактивном тренажере выделяются следующие основные части [14, 24]:

- содержательная часть включает познавательный, демонстрационный компоненты;
- процессуальная часть включает компоненты: моделирующий, контрольный, закрепляющий;
- управляющая часть обеспечивает взаимосвязь между его частями и компонентами;
- диагностическая часть хранит статистическую информацию о работе с конкретными программами.

В процессе разработки и создания электронных учебных ресурсов следует учитывать тот факт, что основной задачей учебной литературы и учебных обучающих средств является преподнесение ученику содержания учебного предмета, объяснение научных принципов материала, выделение основной цели изучения данного материала.

1.3 Анализ существующих веб-тренажеров

Сегодня рынок программных продуктов предлагает большой ассортимент готовых обучающих систем по любой дисциплине, позволяющих обеспечивать весь цикл разработки учебных материалов и управления процессом обучения с использованием дистанционных технологий [5, 8, 10, 23].

Основной акцент делается на использование определенных технических средств и СУБД (Систем Управления Базами Данных), выпускаемых производителями с некоторыми надстройками, обеспечивающими технические возможности построения на их базе процесса дистанционного обучения. Стоимость такого программного обеспечения и его операционные характеристики определяются конкретным заказом.

- 1) Кабинет математики онлайн <http://www.matcabi.net>.

Данная обучающая программа предназначена для помощи в решении различных уравнений, неравенств, матриц и т.п. Также в ней присутствует небольшое количество теории, которую может изучить любой желающий. В теории предоставлена базовая информация и определения из разделов высшей математики. Но главная задача данного тренажера - это решение ряда задач по определенным темам. В тренажере присутствует возможность просмотра пошагового решения заданий, которой возможно использовать после произведения регистрации на данном сайте. Общий внешний вид тренажера, а также все его основные разделы представлены на рисунке 1.2.

MatCabi.net
Кабинет математики онлайн

Математика онлайн Функции Теория **Контрольные/курсовые/рефераты**

Яндекс Виртуальные прогулки по Чернобылю Попробовать

Решение математики онлайн

$\frac{d}{dx}(x^a) = ax^{a-1}$ $\lim_{x \rightarrow 0} \frac{\sin(x)}{x} = 1$

$\sum_{n=0}^{\infty} \frac{1}{n!} = e$ $\int \cos(x) dx = \sin(x) + \text{constant}$

✓ Пошаговое решение на сайте step.matcabi.net

Яндекс Виртуальные прогулки по Чернобылю Попробовать

- Уравнения
- Неравенства
- Матрицы
- Пределы
- Производная
- Сумма ряда
- Неопределенный интеграл
- Определенный интеграл
- Дифференциальные уравнения

Рисунок 1.2 – Общий вид тренажера «кабинет математики онлайн»

2) Математический калькулятор онлайн <http://calc-x.ru>.

Данная программа также предназначена для помощи обучающимся при решении определенных задач. Программа производит онлайн решение различных уравнений, матриц и т.д. Подробное описание решения не предоставляется. Также в данной программе присутствует стандартный калькулятор и небольшое количество теории на определенные темы. Общий

внешний вид приложения, а также все его основные разделы представлены на рисунке 1.3.

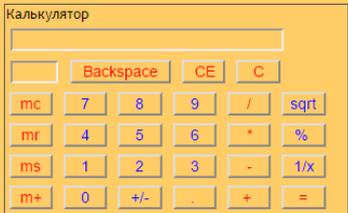
Математический калькулятор онлайн. Математические расчеты, вычисления и решение задач по элементарной и высшей математике на сайте для школы и вуза

На этом сайте имеется математический онлайн сервис, который поможет решить задачи по математике и геометрии, а также поможет при написании дипломных, курсовых и контрольных работ по другим предметам для школы и вуза.

Математический калькулятор на этом сайте выполняет автоматическое и мгновенное решение как простых, так и сложных задач математики, в том числе операции над матрицами, геометрические расчеты, работа с дробями, логарифмами, уравнениями, процентами и т.д. Вы сможете произвести перевод чисел в другую систему счисления и перевод физических величин. Для теоретической помощи существует раздел "Полезное для решения математических задач", в котором можно найти различную табличную и другую информацию. Вычисления доступны 24 часа в сутки с телефона, планшета или компьютера подключенного к Internet. Также вы можете найти онлайн репетитора по математике. Удачной вам учебы, сдачи экзаменов, написании диплома и его защиты!

- [Калькулятор для решения матриц онлайн](#)
- [Геометрия онлайн](#)
- [Другие математические вычисления](#)
- [Полезное для решения математических задач](#)
- [Решение уравнений онлайн](#)
- [Дискретная математика онлайн](#)

- [Преобразование и другие операции над матрицами. Калькулятор для решения матриц онлайн](#)
 - [Умножение матриц онлайн](#)
 - [Онлайн вычитание матриц](#)
 - [Онлайн транспонирование матриц](#)
 - [Онлайн возведение матриц в степень](#)
 - [Онлайн сложение матриц](#)
 - [Умножение матрицы на число онлайн](#)
 - [Онлайн вычисление определителя матрицы \(найти детерминант\)](#)
 - [Онлайн вычисление ранга матрицы \(найти ранг матрицы\)](#)
 - [Онлайн приведение матриц к треугольному виду](#)
 - [Нахождение обратной матрицы онлайн](#)
- [Геометрия онлайн. Решение задач по геометрии в реальном времени](#)
 - [Вычисление площади фигур онлайн](#)
 - [Вычисление объема фигур онлайн](#)
 - [Вычисление периметра фигур онлайн](#)
 - [Вычисление длины и расстояния онлайн](#)



Калькулятор

Buttons: Backspace, CE, C, mc, 7, 8, 9, /, sqrt, mr, 4, 5, 6, *, %, ms, 1, 2, 3, -, 1/x, m+, 0, +/-, ., +, =

Рисунок 1.3 – Общий вид тренажера «математический калькулятор онлайн»

В таблице 1.1 приведен сравнительный анализ рассмотренных веб-тренажеров по сформулированным критериям.

Таблица 1.1 – Сравнительный анализ аналогов

		Кабинет математики онлайн http://www.matcabi.net	Математический калькулятор онлайн http://calc-x.ru
1	Работа с преподавателем	-	-
2	Онлайн решение необходимых задач	+	+
3	Подробное объяснение решенных задач	+	-
4	Наличие теоретического материала	+	+

Продолжение таблицы 1.1

		Кабинет математики онлайн http://www.matcabi.net	Математический калькулятор онлайн http://calc-x.ru
5	Вводное тестирование для определения уровня знаний	-	-
6	Итоговое тестирование	-	-
7	Регистрация	+	-
8	Бесплатные услуги	-	+
	Итого	4/8	3/8

Исходя из анализа рассмотренных подобных системы, могут быть определены основные функциональные требования к разрабатываемому программному продукту: работа с преподавателем; онлайн решение необходимых задач; подробное объяснение решенных задач; наличие теоретического материала; вводное тестирование для определения уровня знаний; итоговое тестирование; регистрация.

Таким образом, определены основные функциональные требования, для решения которых необходимо определиться с аппаратными возможностями и структурными компонентами.

1.4 Требования к разрабатываемому тренажеру

Веб-тренажер определяется как учебное средство, предназначенное для формирования и совершенствования у обучаемых профессиональных навыков и необходимых им умений путем многократного выполнения обучаемыми действий [25].

Проектируемый веб-тренажер должен обладать высоким качеством изображения [26, 28]. В виртуальном тренажере должно быть предусмотрено два режима работы: режим «Тренировки» и режим «Выполнения работы». В

режиме «Тренировки» студенту будет предложено ознакомиться со всеми этапами работы: просмотром изучаемого материала, закреплении его путем решения подобных задач и прохождением тестирования для проверки уровня знаний усвоенного материала. Режим «Тренировки», по сути, является работой по алгоритму, который должен быть представлен не в виде непрерывного видео или анимационного клипа, а как последовательность шагов. Представление тренировочного режима веб-тренажера в виде алгоритма обеспечивает повышение гибкости взаимодействия его с пользователем, предоставляя возможность неоднократного просмотра отдельных модулей тренажера в любой последовательности. Проверки действий пользователя в режим «Тренировка» не вводятся.

В режиме «Выполнения работы» студенту предлагается воспроизвести ранее изученную последовательность действий, при этом существует система контроля, следящая за правильностью действий пользователя.

Веб-тренажер как средство обучения должен предоставлять пользователям следующие функциональные возможности [22, 25]:

- подготовка исходного материала для обучения, а именно:
 - разработка базы данных параметров, соответствующих задачам обучения;
 - разработка видеоклипов для представления информации;
 - подготовка набора различных задач и реализация их с помощью той или иной математической модели, а также информации для контроля правильности действий или ответов обучаемого;
 - контекстное представление теоретического материала;
- регистрация и ведение базы данных обучаемых, включающей статистическую информацию о прохождении процесса обучения:
 - проведение собственно процесса обучения;
 - предоставление возможности анализа статистики прохождения процесса обучения с целью как оценки успеваемости обучаемых, так и оценки необходимости корректировки учебного материала.

Основной задачей интерактивных компьютерных тренажеров является привитие навыков определенного вида практической деятельности. Таким образом, перед тренажером стоит задача: научить решать математические примеры различного типа, освоив работу численных алгоритмов и т.д.

Решение любого типа задачи может быть описано с помощью некоего алгоритма как последовательности вычислительных и других действий, однозначно приводящих к ответу на поставленную задачу, исходя из некоторого набора известных и заданных данных [3, 10].

Исходя из определения алгоритма как последовательности шагов для достижения поставленной цели, ход решения можно представить в виде направленного графа, изображенного в виде дерева, в котором поддеревья являются подзадачами (И-Деревья в системном анализе); листья – элементами списка (последовательности шагов, описывающих алгоритм); а ребра, присутствующие в графе, но отсутствующие в дереве, задаются листьями с командой типа «перейти на пункт N» (рис. 1.4) [2].

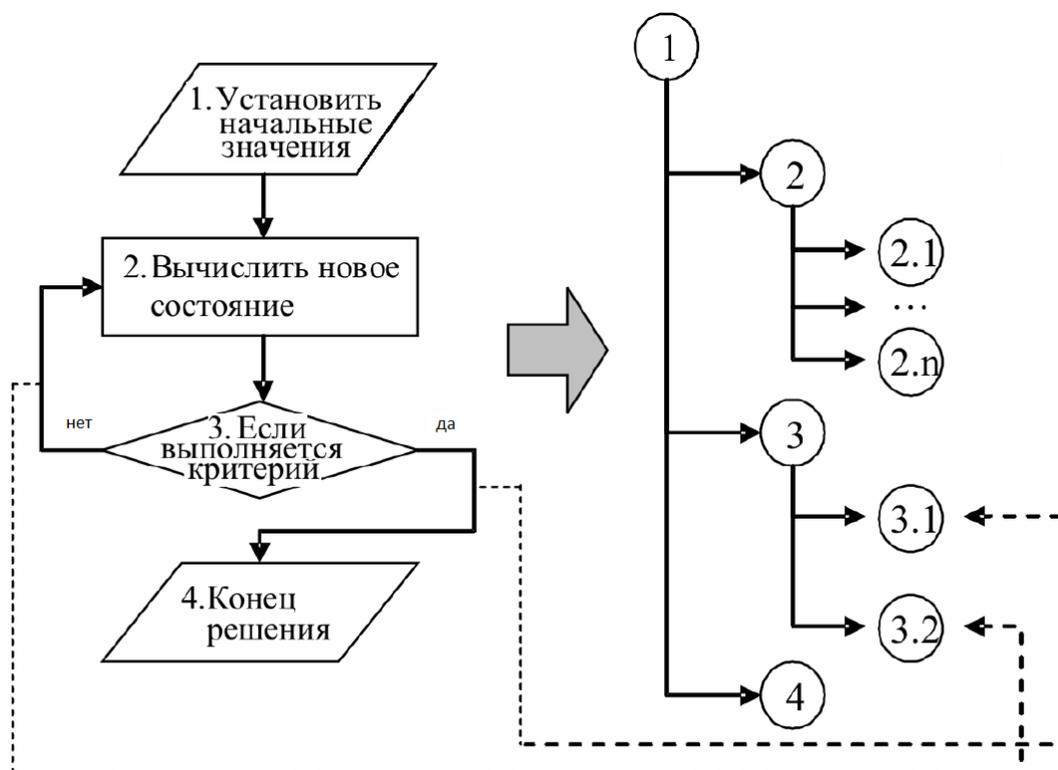


Рисунок 1.4 – Варианты представления алгоритма

Инструментами отображения алгоритма являются:

- блок-схема алгоритма;

- компонент наглядного графического представления работы алгоритма (выраженного в виде графа).

Разработка тренажера будет осуществляться по принципу сценария. Сценарий явно определяет роли сторон и последовательность действий, следовательно, и алгоритм решения задачи и алгоритм работы с тренажером укладываются в рамки этого понятия.

Сценарий веб-тренажера – система моделей, полно и однозначно описывающих его работу [4, 5]. Данная система содержит в себе описание логического пути программы и формальное описание последовательности действий при решении задач.

Разберем структуры данных, предназначенных для раскрытия сформированного сценария занятия по практике, веб-тренажера. Для начала задается траектория хода работы с веб-тренажером. При этом перед решением задачи возникает необходимость изучения справочного материала по определенной дисциплине. Таким образом первой структурой будет считаться, последовательность действий при работе с веб-тренажером. Древовидная структура в данной ситуации является наиболее приемлемой. В данной структуре «листья» описывают (идентифицируют) дидактические единицы или программные формы, предоставляемые пользователю на отдельных этапах работы с веб-тренажером; и описывают отсутствующие в классическом дереве ребра графа переходов.

Элементом сценария будет являться описание работы алгоритма решения задачи. Он содержит в себе древовидную структуру алгоритма, описание структуры фазовой переменной алгоритма и функциональную таблицу смежности дерева алгоритма, которая предоставляет информацию об переменных в фазовом состоянии алгоритма при переходе от вершины к вершине.

Веб-тренажер делится на две основные подсистемы [4, 25]: разработки и использования сценариев. Подсистема разработки сценариев и ресурсов для них состоит из:

- модуля для описания дерева алгоритма работы с веб-тренажером;
- модуля для описания дерева алгоритма решения задач.

Подсистема использования сценариев состоит из модуля реализации сценариев и всех сопряженных с ним ресурсов.

В общем случае веб-тренажер предполагается строить на принципе «клиент-сервер» с обеспечением многопользовательского доступа.

Вывод по главе

В первой главе были рассмотрены сущность, характеристики структуры, особенности и основные этапы создания веб-тренажера по дисциплине.

В результате анализа основных подходов к реализации веб-тренажеров, а также анализа аналогичных обучающих программ можно сделать вывод и выделить определенные критерии, необходимые любой обучающей системе, в частности веб-тренажеру: регистрация (для индивидуального учета знаний обучаемого), консультация с учителем.

Глава 2 Моделирование веб-тренажера по дисциплине «Теория вероятностей и математическая статистика»

2.1 Моделирование архитектуры обучающей системы «Веб-тренажер»

При разработке компьютерного обучающего веб-тренажера берется во внимание разработка архитектуры веб-тренажера.

Понятие архитектуры системы (структуры системы) заключается в разделении каждой программы на три составляющих [2, 4]:

- представление данных;
- прикладная логика;
- управление базой данных.

Функция, предназначенная для обработки данных, появляется на нескольких серверах, что говорит об многоуровневой клиент-серверной архитектуре. В данных системах функции сбора, хранения, обработки и подачи информации разделяются, что позволяет использовать вычислительные ресурсы серверов и клиентов эффективней.

Многоуровневая архитектура тренажера изображена на рисунке 2.1. Модель трехуровневой клиент-серверной архитектуры позволяет разделить функционал обучающего тренажера по трем различным уровням [11, 12]:

- клиента,
- сервера приложений,
- сервера данных.

Все подсистемы веб-тренажера соответствуют своим функциональными требованиям. Это обеспечивает индивидуальность материалов для различных пользователей веб-тренажера.

Права доступа - это действия, разрешенные пользователям для выполнения работ над объектами данных. Одним из главных принципов контроля доступа в веб-тренажере является выборочное управление доступом.

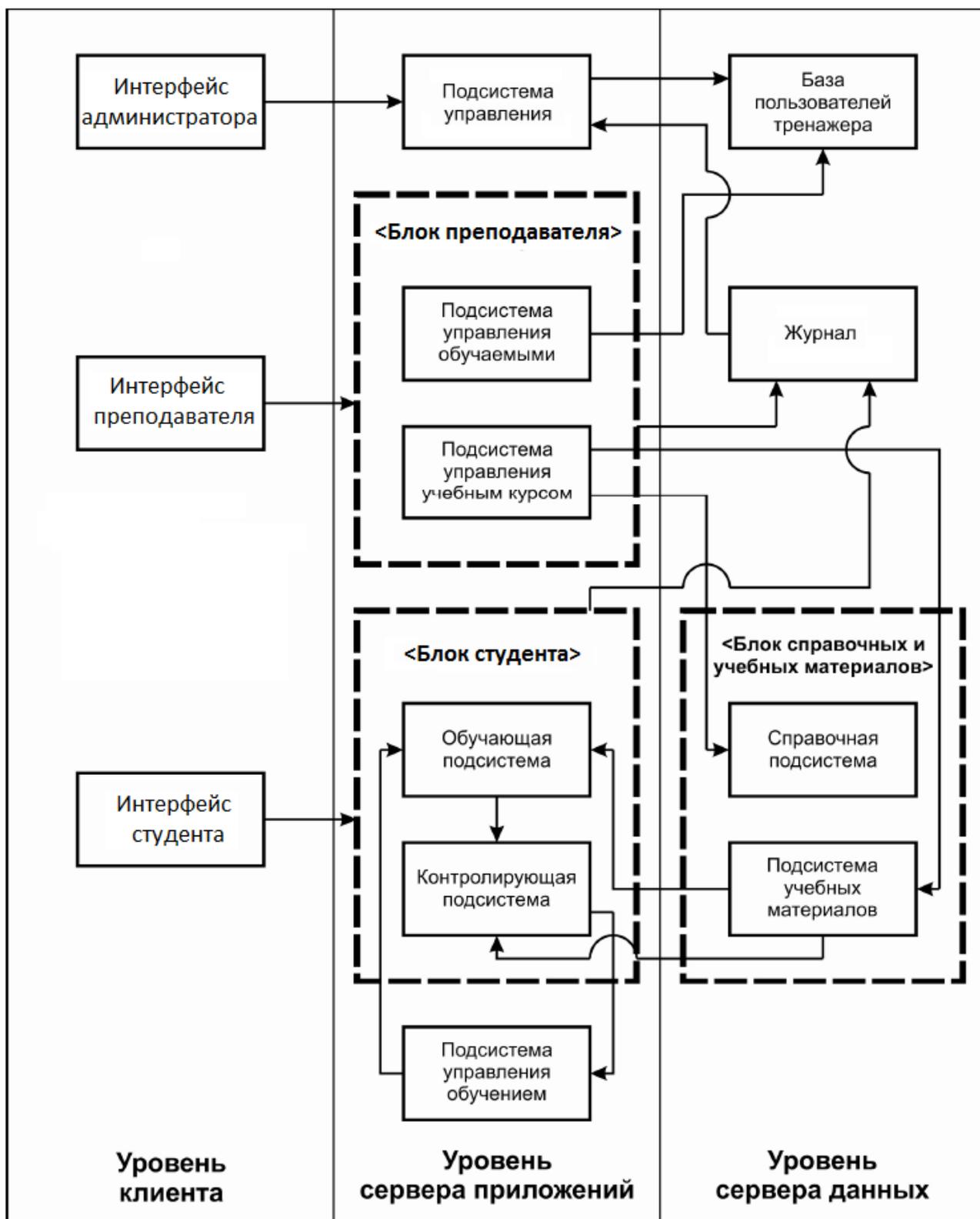


Рисунок 2.1 – Клиент-серверная архитектура веб-тренажера

Основные пользователи такой системы разделяются на три группы: администратор, преподаватели и студенты. Для каждой группы интерфейс уникален и обладает своей функциональной задачей. Интерфейсы располагаются на уровне клиента, это значит, что программные модули

находятся на разных компьютерах. На втором уровне - уровне сервера приложений реализуется основная бизнес-логика тренажера для подготовки студентов, а на третьем уровне сервера данных находится главное хранилище файлов и СУБД [12, 27].

Только администратором тренажера осуществляется доступ к подсистеме управления преподавателями. Студенты в тренажер добавляются самостоятельно, с помощью регистрации. В базе хранятся профили всех студентов. Кроме администрирования и консультации студентов, в задачи преподавателя также входит создание и поддержка самого учебного курса тренажера, который выражается в представлении различных тем, структурированных по уровню сложности.

Подсистема управления обучением способствует настройке хранимых на сервере данных справочной подсистемы и подсистемы учебных материалов. Справочная подсистема включает в себя основные методические инструкции по работе с учебным курсом и веб-тренажером в целом и размещается в базе данных. В файловом хранилище находятся разные документы в том числе и визуализированные модели, а характеристики обучающих и контролирующих подсистем – в базе данных. Все данные хранимые в подсистеме учебных материалов ни раз используются в дальнейшем при обучении, проходящем на сервере приложения.

Работа студента в тренажере начинается с прохождения входного контроля. При первичной проверке анализируются общие теоретические и практические навыки выполнения работ. Исходя из итогов входных данных подсистема управления обучением составляет определенную последовательность изучения материалов обучаемым, другими словами создает для него индивидуальный образовательный курс.

После прохождения индивидуального процесса обучения, студент переходит в контролируемую подсистему, в которой проходит итоговая проверка его умений, знаний и навыков. Если обучающийся получает неудовлетворительные результаты после итоговой проверки, индивидуальный

курс обучения корректируется с учетом ошибок, и цикл обучения повторяется заново. Все действия преподавателей и студентов записываются в журнал.

Таким образом, была определена архитектура программного приложения, а также общий сценарий его работы.

2.2 Функциональное моделирование обучающей системы «Веб-тренажер»

Сценарий работы компьютерного тренажера продемонстрирован на диаграмме прецедентов (рис. 2.2). На данной диаграмме рассмотрен процесс управления студентами. Помимо регистрации преподаватель указывает доступные темы учебного курса, а также во время всего процесса обучения он имеет доступ к подробной статистике и итогам обучения каждого студента. Студент же может не получить необходимые знания за один сеанс работы с тренажером, поэтому все действия студентов в веб-тренажере фиксируется и заносится в базу данных [1, 2, 11].

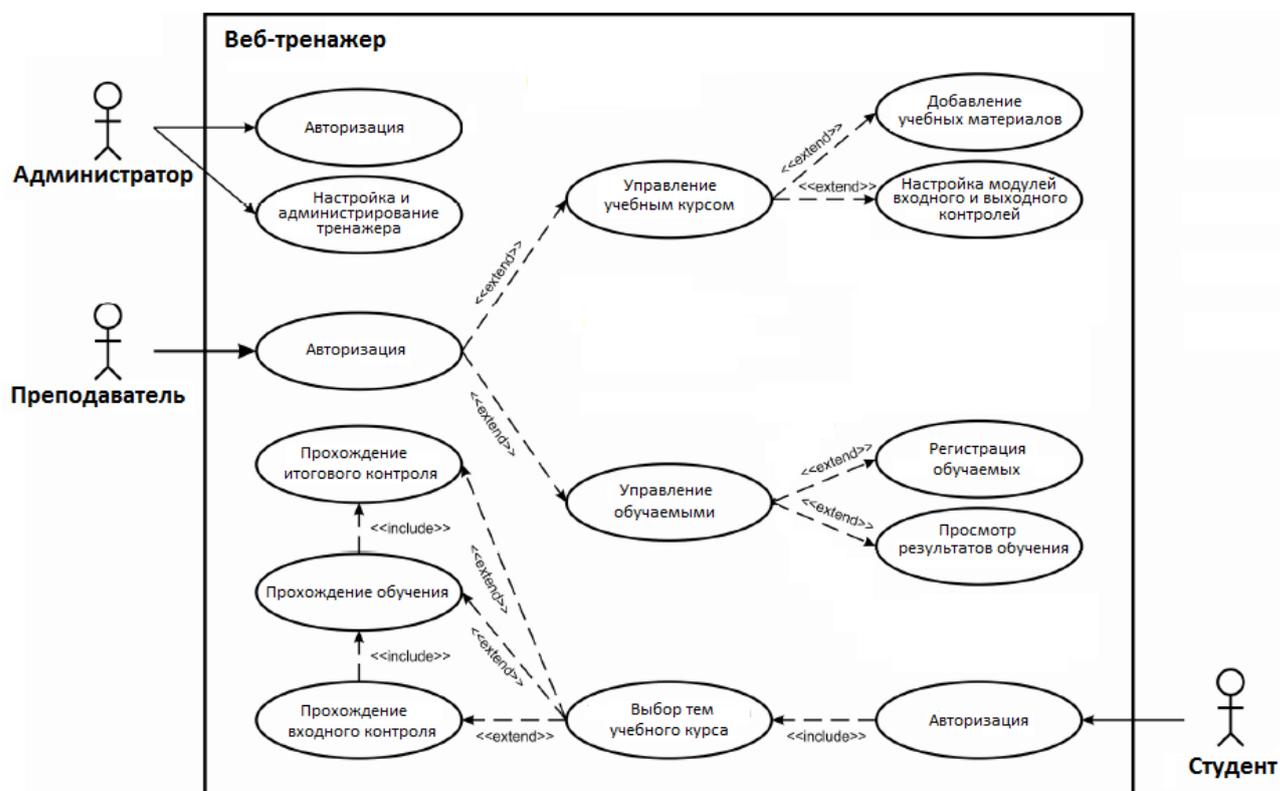


Рисунок 2.2 – Диаграмма прецедентов веб-тренажера

Создание обучающей системы, реализованной в виде веб-тренажера с использованием инструментальных программ, должна пройти четыре периода:

1. Написание сценария веб-тренажера: нужно продумать материал обучающей программы, а также вопросы для тестирований и ответы на них.
2. Ввод данных по обучающим текстам, формирование контролирующих фрагментов: вопросов, ответов на них и способов нахождения верных ответов.
3. Сопряжение между собой определенных элементов обучающей программы в единую диалоговую систему, установление смысловых связей между темами, вопросами и помощью.
4. Контроль программы во время ее использования и исправление недостатков.

На основе анализа данной диаграммы были определены основные функции основных пользователей системы и выделены основные ее модули:

1. Модуль администратора – данный модуль предназначен для управления веб-тренажером. Данный модуль позволит управлять существующими пользователями системы.
2. Модуль преподавателя – данный модуль предназначен для преподавателей. Данный модуль позволит управлять учебными материалами и тестовыми заданиями, при необходимости добавляя или убирая необходимые курсы или информацию. А также наблюдать статистические данные о успеваемости студентов.
3. Модуль студента – данный модуль предназначен для студентов. Данный модуль позволит студентам получать доступ к учебным материалам и выполнять тестовые задания, тем самым проверяя и корректируя свои знания по данной дисциплине.

Таким образом, были сформулированы основные функциональные требования к каждой группе пользователей, для которых предназначается веб-тренажер.

2.3 Концептуальная модель веб-тренажера

Для описания работы веб-тренажера необходимо построить модель. Такая модель должна быть адекватна предметной области; следовательно, она должна содержать в себе знания всех участников работы с тренажером.

Диаграмма потоков данных (data flow diagram, DFD) – является одним из инструментов анализа структуры и реализации информационных систем, существующим достаточно долго [1, 11].

Модель DFD – является иерархической моделью. Ко всем ее процессам может быть применена декомпозиция, то есть разбиение целого на структурные составляющие, отношения которых могут быть отображены на отдельной диаграмме [1, 2]. При достижении требуемой глубины декомпозиции – в процессе нижнего уровня появляется текстовое описание (мини-спецификация).

Схема потоков данных в реализуемом веб-тренажере представлена на рисунке 2.3.

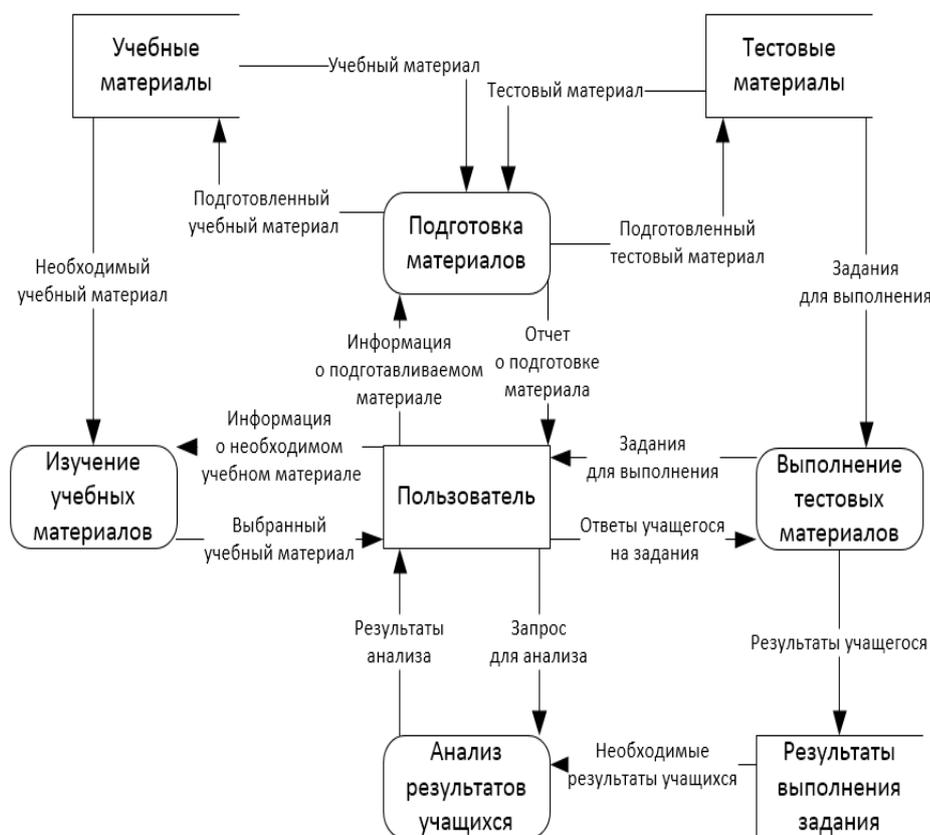


Рисунок 2.3 – Диаграмма потоков данных веб-тренажера

В данной модели у пользователя есть выбор:

1. В режиме преподавателя:
 1. Подготовить учебные материалы.
 2. Выполнить анализ результатов учащихся.
2. В режиме студента:
 1. Изучить учебные материалы.
 2. Выполнить тестовые задания.

Далее рассмотрим подробнее процесс подготовки учебных материалов (рис. 2.4).

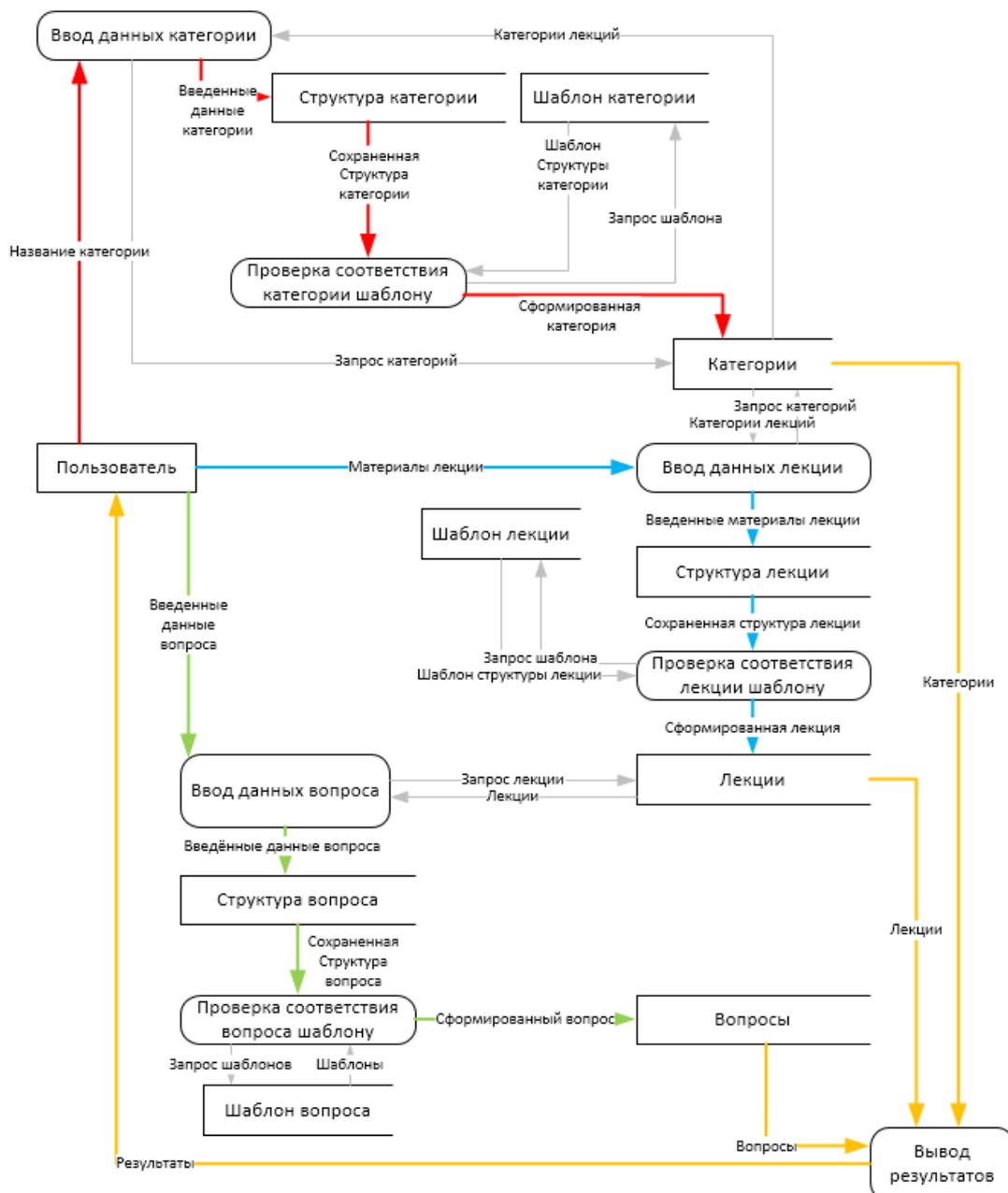


Рисунок 2.4 - Модель процесса создания учебных материалов

С самого начала у пользователя есть выбор:

1. Ввести данные темы, выполнить проверку введенных данных и получить результаты создания тематической категории.
2. Ввести данные лекции, выполнить проверку введенных данных и получить результаты создания лекции.
3. Ввести данные вопроса, выполнить проверку введенных данных и получить результаты создания вопроса к лекции.

Описав общее представление веб-тренажера, перейдем к описанию структуры базы данных, используемой в нем.

2.4 Моделирование базы данных веб-тренажера

База данных – это целевая модель предметной области. Процесс проектирования базы данных начинается с установления концептуальных требований пользователей системы [12, 27]. Обычно концептуальная модель базы данных состоит из:

1. Описания понятий предметной области (информационных объектов) и связей между ними.
2. Описание ограничений целостности (требований к допустимым значениям данных и к их соотношению).

Инфологическую модель можно назвать описанием предметной области, появившемся благодаря анализу семантики объектов и явлений. Оно реализуется при отсутствии ориентации на использование программных или технических компьютерных средств [27, 30].

Наиболее широко используемым средством разработки логических моделей баз данных являются диаграммы «сущность-связь» – Entity-Relationship (ER-диаграммы). Диаграмма даталогической модели данных, созданная при помощи свободно распространяемого программного обеспечения «DBDesigner 4», представлена на рисунке 2.5.

В представленной модели все сущности имеют связь «1:М». Модель приведена к третьей нормальной форме.

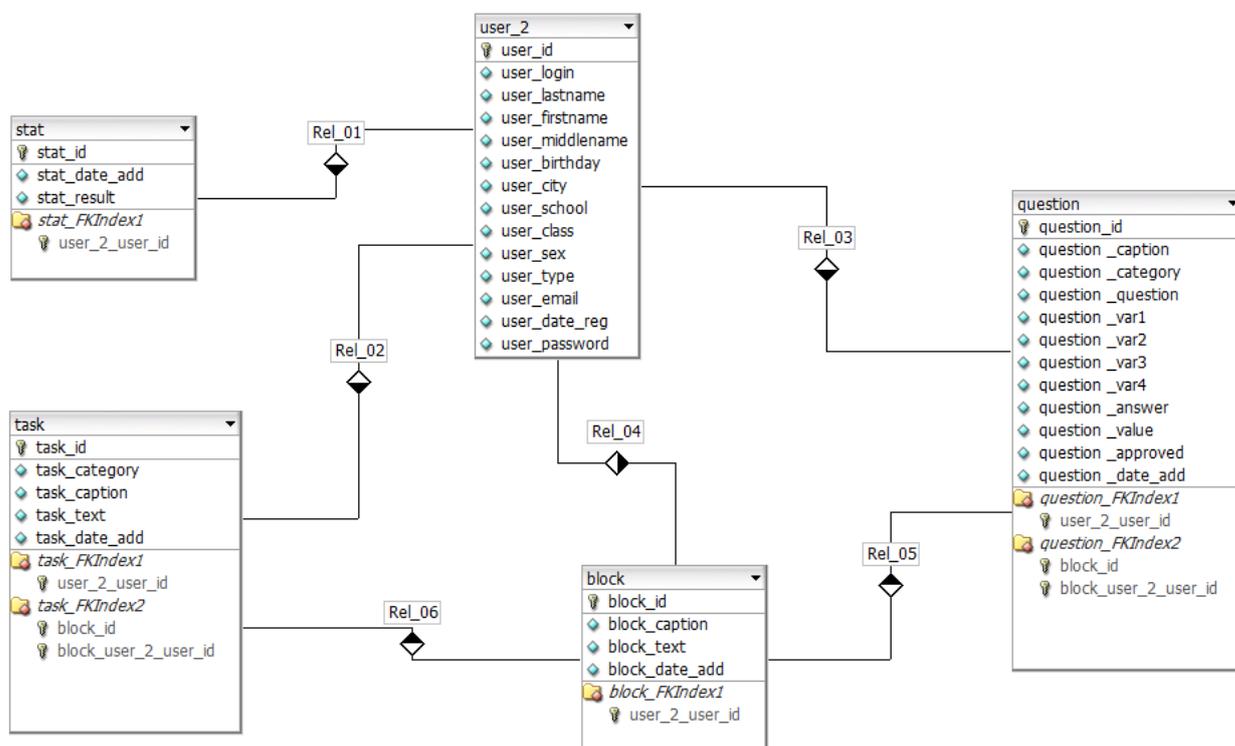


Рисунок 2.5 - Даталогическая модель данных веб-тренажера

После создания логической модели переходим к принятию решения о выборе платформы реализации.

Выбор платформы зависит от требований к использованию данных и стратегических принципов формирования архитектуры системы.

Компонентами физической модели данных являются таблицы, столбцы и отношения. Сущности логической модели, вероятно, станут таблицами в физической модели. Логические атрибуты станут столбцами. Логические отношения станут ограничениями целостности связей. Физическая модель зависит от платформы, выбранной для реализации, и требований к использованию данных [27].

Схематично физическая модель данных представлена на рисунке 2.6.

При разработке схемы, была использована программа «DBDesigner 4» с включенным параметром «Physical Schema Level». Это делает возможным просмотреть типы данных для каждой таблицы.

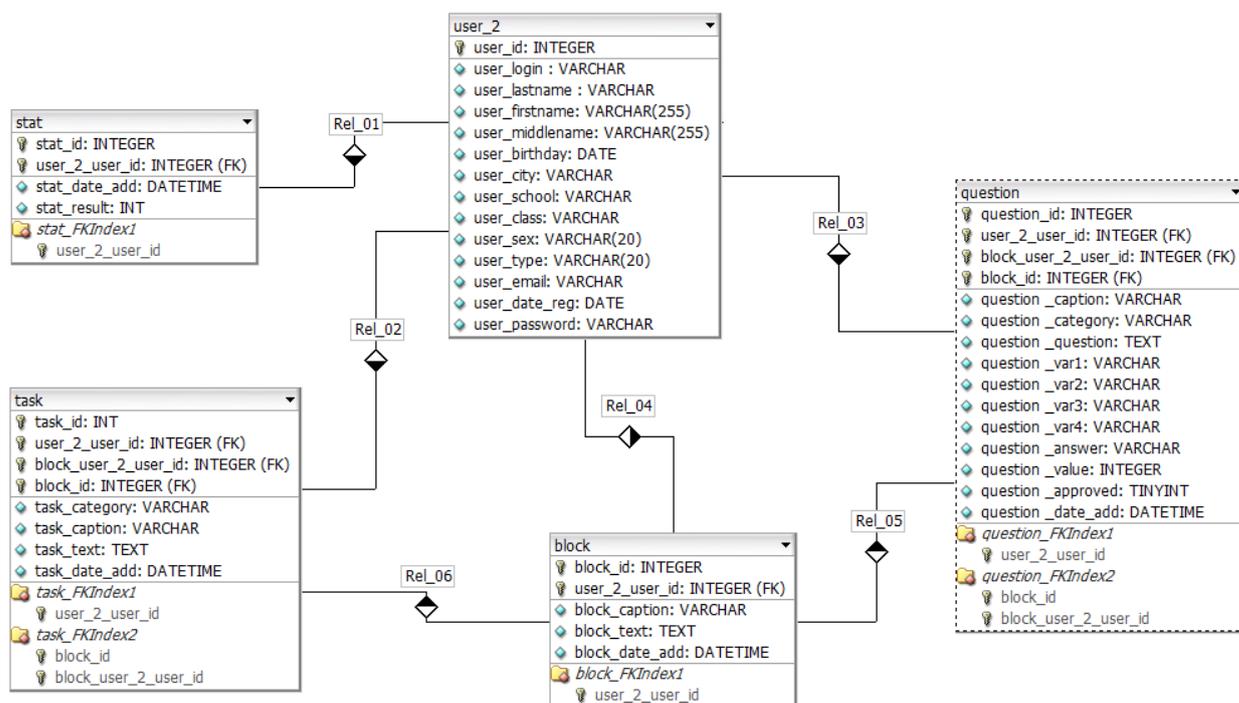


Рисунок 2.6 - Физическая модель данных веб-тренажера

В заключении отметим, что разработанная модель базы данных предоставляет возможность реализовать все модули и функции обучающей системы.

2.5 Математическая модель обучающей системы «Веб-тренажер»

Математическую модель тренажера можно условно разделить на две модели – модель функционирования тренажера и модель представления работы какого-либо изучаемого алгоритма [6].

Работа тренажера описывается конечным детерминированным автоматом с памятью: $M_1(X_1, S_1, V_1, \varphi, \psi)$ (рис. 2.7). Здесь:

- X_1 – входной алфавит, элементы которого задают управляющее воздействие со стороны пользователя;
- S_1 – алфавит состояний, элементы которого определяют текущий фрейм тренажера;
- V_1 – выходной алфавит, элементы которого определяют записи в протоколе работы пользователя с тренажером;
- φ – выходная функция времени (для формирования штампа времени в файле протокола) и состояния, задаваемая при помощи таблицы;

- ψ – переходная функция, определяющая логику функционирования самого тренажера, задается таблично.

$s \backslash x$	x_1	\dots	x_j	\dots	x_k
s_0					
s_1					
s_i			$\delta(s_i, x_j)$		
			$\lambda(s_i, x_j)$		
s_{r-1}					

Рисунок 2.7 – Представление конечного автомата, положенного в основу описания веб-тренажера

Выходные состояния $b \in B_1$ зависят от текущего состояния $s \in S_1$ и входного символа вола $a \in X_1$, т.е.

$$b = \varphi(a, s) \quad (1)$$

Аналогично следующее состояние автомата также зависит от текущего состояния и входного символа и задается переходной функцией:

$$s' = \psi(a, s) \quad (2)$$

Работа изучаемого математического алгоритма описывается с помощью детерминированного бесконечного автомата: $M_2(X_2, S_2, B_2, \varphi, \psi)$. Здесь:

- X_2 – входной алфавит. Все его элементы определяют большое количество возможных значений фазовой переменной алгоритма;
- S_2 – алфавит состояний. Он анализирует состояние алгоритма, соответствующее текущему шагу его выполнения;
- B_2 – выходной алфавит. Аналогичен входному;
- φ – выходная функция. С помощью неё определяются изменения фазового состояния алгоритма в соответствии с логикой его работы, задается таблично;

- ψ – переходная функция. Данная функция определяет логику работы алгоритма, задается графом переходов.

Формирование работы алгоритма показано на рисунке 2.8.

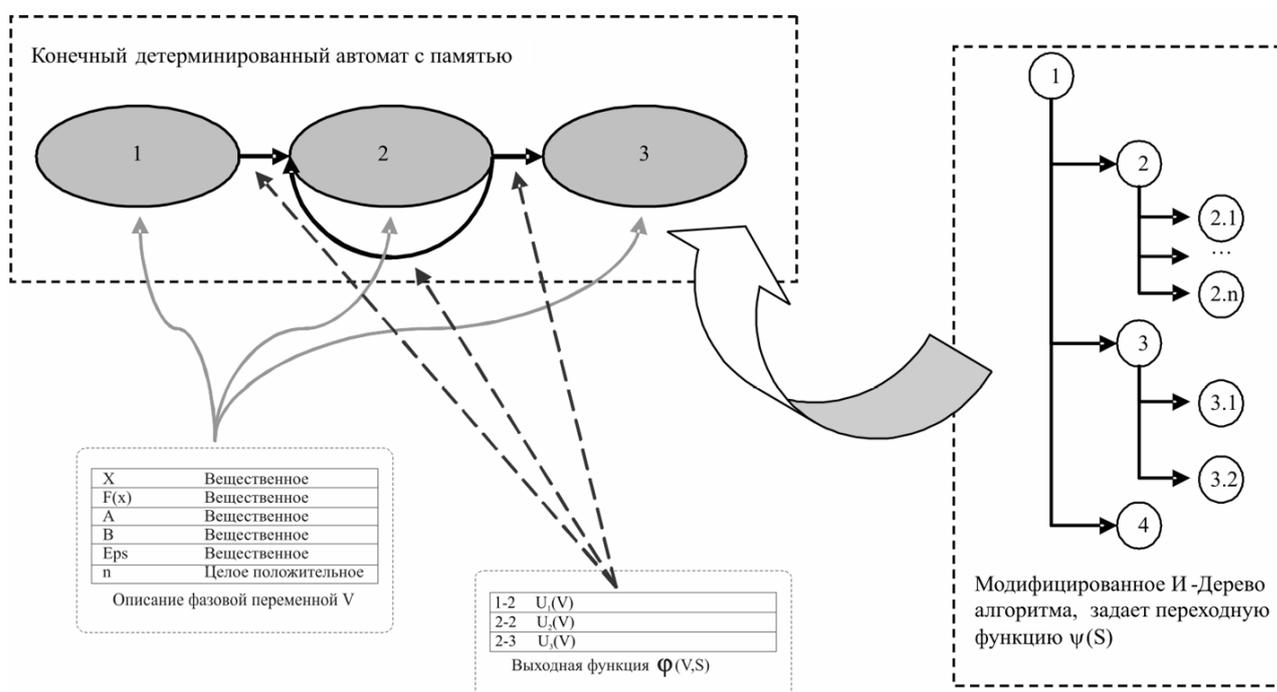


Рисунок 2.8 - Формирование модели работы алгоритма

Структуры, определяющие суть фазовой переменной, выходной функции и модифицированного И-Дерево алгоритма, создаются при помощи: модуля описания дерева алгоритма решения задачи; модуля описания функциональной таблицы смежности для дерева алгоритма решения задачи и модуля описания фазовой переменной алгоритма [6].

Введем следующие обозначения: M_0 - модель предметной области; $M_L = M_L(S)$ - модель ученика, где S - обозначается вектор параметров определенной модели; $V = (v_1, v_2, \dots, v_k) \subseteq D_v$ - коллекция контрольных заданий.

Основой конечно-автоматной модели обучения являются автоматные модели студента M_L и обучающей программы (преподавателя) M_T . Выходом X автомата M_T (и входом автомата M_L) являются учебный материал и контрольные задания. Выходом Y автомата M_L (и входом автомата M_T) считаются события, произошедшие в процессе обучения.

Задачей веб-тренажера (управляющего автомата M_T) является управление

при помощи выхода X автоматом M_L так, что он приводится в назначенное состояние (достигает заданного учебного результата) за минимальное время. Задача состоит также в достижении максимально возможного результата за заданное время или в уменьшении числа ошибок на выходе Y автомата M_L на заданном интервале времени.

В любом случае управление обучением, реализованном в рамках использования веб-тренажера, разбивается на следующие этапы:

- 1) предоставление студенту какой-либо части обучающего материала;
- 2) протоколирование процесса прохождения учебного материала;
- 3) анализ текущей ситуации;
- 4) корректировка учебных заданий.

Следовательно, автомат M_T может быть представлен в виде суперпозиции автоматов $M_T^1, M_T^2, \dots, M_T^4$.

Автомат M_T^1 является моделью интерфейса студент – веб-тренажер. В простейшем случае данный автомат может быть представлен в виде автомата с единственным состоянием.

Автомат M_T^2 отвечает за написание протокола обучения. Множество состояний данного автомата является декартовым произведением множеств $S_{2,1}$, $S_{2,2}$ и $S_{2,3}$, характеризующих протокол обучения, модель студента и модель учебной ситуации соответственно.

Автомат M_T^3 является главным в модели процесса обучения. Задача данного автомата - принятие решений о действиях, которые следует выполнить в конкретной сложившейся ситуации – предоставить ученику контрольное задание, предоставить ему учебный материал пройденного им урока повторно или же вернуться в начало изучаемой темы и т.п.

Для реализации заданной задачи можно воспользоваться системами продукций. В их левой части расположены условия на значения входа, а в правой – выхода. Но более предпочтительным, все же, будет решение, построенное на стереотипирование ученика.

Предполагается, что автомат M_L является вероятностным автоматом из некоторого класса $M_L = (M_L^1, M_L^2, \dots, M_L^k)$, где все подклассы M_L^i соответствует одному из типов ученика. В зависимости от состояния автомата M_T^2 автомат M_L относится к одному из подклассов M_L^i . В дальнейшем, зная этот подкласс, автомат M_T^3 составляет определенную последовательность действий, переводящих автомат M_L в более подходящее состояние для конкретной учебной ситуации.

Автомат M_T^4 занимается управлением базой данных учебных стратегий, для которой запросом является вход, а выход будет являться результат обработки данного запроса. Данный автомат также может быть реализован как как автомат управления базой данных учебных материалов, задача которой – подобрать материал, отвечающий на требуемый запрос.

Таким образом, веб-тренажер можно представить, как конечный автомат, для которого определены входные и выходные параметры его функционирования.

2.6 Структурная схема представления веб-тренажера

Структура интерактивного веб-тренажера определяется тем, что веб-тренажеры и прочие обучающие системы предназначены для организации самообучения и должны четко разбираться, в каких разделах и в каком порядке они должны быть изучены и как связаны между собой. Также должна быть распределена последовательность изучаемого материала: теоретическая и практическая части, контрольные задания, демонстрации и материалы для дополнительного изучения [5].

Любому веб-тренажер необходимо включать в себя данные компоненты:

- средства изучения теоретического материала по теме;
- средства поддержки практикумов и лабораторных работ;
- средства контроля знаний в виде тестовых систем;
- средства диалога между учителем и студентом;

- методические комментарии с рекомендациями по изучению темы;
- средства управления процессом обучения.

На рисунке 2.9 представлена структура интерактивного веб-тренажера.



Рисунок 2.9 – Структура веб-тренажера

Веб-тренажер предназначен, прежде всего, для дистанционной формы обучения. Поэтому он, кроме основного текста, содержит справочный материал, необходимый при возникновении затруднений во время самостоятельной работы. Поэтому для данного веб-тренажера важно наличие гиперссылок, включающие в себя указатели, дополнительный текст, мультимедиа, списки определений.

Вывод по главе

Таким образом, ознакомившись со всеми структурами, математической и физической моделью и объяснив в общих чертах, что же все-таки необходимо для современного веб-тренажера, можно сделать вывод, что веб-тренажер должен самостоятельно функционировать и определять уровень знаний обучающегося, а также содержать все справки и инструкции по своему использованию.

Глава 3 Реализация веб-тренажера по дисциплине «Теория вероятностей и математическая статистика»

3.1 Функциональное моделирование веб-тренажера как средства обучения дисциплине

Веб-тренажер – представляет собой совокупность теоретического материала по теме и практического применения полученных знаний посредством теста.

Проектируя обучающую систему, реализованную в виде веб-тренажера, были определены основные функции и модули системы.

Данная система представлена совокупностью модулей, которые представляют собой: модуль «Администратор», модуль «Преподаватель», модуль «Ученик».

Проектируя обучающую систему, было определено, что для каждого пользователя определены права [4, 5]. Рассмотрим, какие действия могут совершать различные пользователи системы:

1. Администратор – это пользователь системы с расширенными правами. Возможности, присущие данному пользователю: редактирование прав других пользователей системы, удаление других пользователей.

2. Студент – пользователь с правами учащегося. Возможности, присущие данному пользователю: просмотр изучаемого материала, прохождение тестирования, осуществление выхода из системы.

3. Преподаватель – пользователь с правами учителя. Возможности, присущие данному пользователю: все действия, доступные пользователю «Студент»; добавление, редактирование и удаление лекционных материалов; добавление, редактирование и удаление тестовых заданий; проверка тестовых заданий и просмотр статистики.

Схема с изображением функциональных модулей системы изображена на рисунке 3.1.

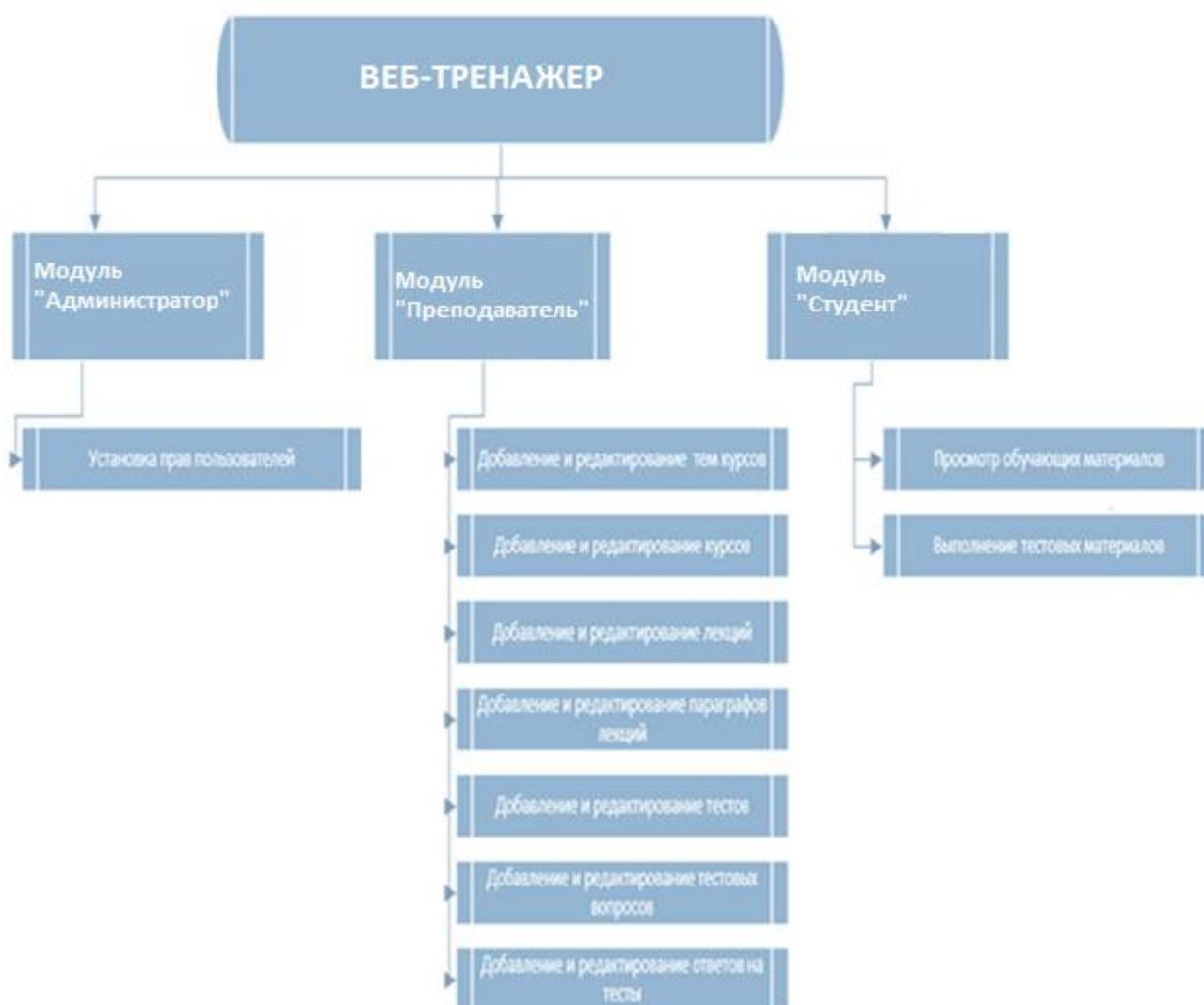


Рисунок 3.1 – Функциональная схема веб-тренажера

На функциональной схеме изображены все модули системы, а также функции, которые они выполняют. В основании всей системы лежит главный модуль, который обеспечивают взаимодействие со всеми функциональными модулями системы.

Структура веб-тренажера как обучающей системы включает в себя определенные функциональные элементы и различное программное обеспечение, необходимое для выполнения основной целевой установки - обеспечение процесса обучения студентов по темам учебного курса.

Структурная схема веб-тренажера изображена на рисунке 3.2 и включает в себя следующие подсистемы: теории, практики, тестирования и оценки знаний студентов.



Рисунок 3.2 - Структура веб-тренажера, представленной в виде модулей

Подсистема теории должна довести до обучаемого теоретическую информацию, предусмотренными учебным курсом. Теоретическая информация предоставляется темами, отображенными на страницах в виде, легко воспринимаемом с экрана дисплея. Это дает возможность просмотреть всю информацию, полностью ее усвоить, переработать и перейти к следующей. При необходимости предусмотрена возможность возврата к предыдущим темам курса. Большая часть лекционного материала иллюстрируется примерами. Данная подсистема позволяет осуществить выбор нужной для студента темы для более детальной проработки материала, который вызывает наибольшие затруднения.

Подсистема практики предназначена для наглядного показа студентам выполнения практических примеров и проверки усвоения принципов их решения. Подсистема практики обеспечивает наглядность самого процесса обучения, практически без непосредственного участия студента.

Подсистема тестирования обеспечивает возможность задания набора контрольных вопросов для проверки теоретических знаний по ранее изученным темам, а также возможность возврата к повторному изучению теории.

Подсистема оценки знаний обеспечивает анализ результатов работы студента с веб-тренажером, а также сообщает окончательную оценку за сеанс работы. Она предназначена для: контроля выполнения практических примеров, выставления оценки и возврата к тренирующей части при неудовлетворительных результатах. Данная подсистема построена по разветвленному алгоритму анализа ответов, который обеспечивает взаимодействие с подсистемой теории и практики для полного понимания материала и устарения возникающих при обучении ошибок (т.к. позволяет заново просматривать как теоретический материал, так и примеры решения практических задач). Для формирования оценки используется комплексный метод анализа сделанных ошибок, в зависимости от установления фактов наличия ошибок и времени решения.

Таким образом, определена модульная структура построения веб-тренажера, что предъявляет особые требования к выбору средств его реализации.

3.2 Выбор средств реализации интерактивного веб-тренажера

Состав используемого программного обеспечения определяется прежде всего тем составом подсистем, которые будут входить в реализуемый веб-тренажер. Поэтому рассмотрим наиболее употребляемые языки программирования, при помощи которых можно реализовать проектируемую обучающую систему [5, 8, 12].

Java – объектно-ориентированный язык программирования. Главным достоинством языка Java является то, что его приложения после компиляции в специальный байт-код, выполняются в виртуальной машине в любой операционной системе и на любом оборудовании. Также немаловажным фактором является достаточно высокая безопасность языка. При любой попытке неразрешенного доступа или соединения программа прерывает свою работу, потому что все приложения контролируются Java-машиной. Помимо

этого, существует огромное количество бесплатных программных библиотек к этому языку.

Недостатком языка является чрезмерная нагрузка на оперативную память оборудования. Также время выполнения одних и тех же задач в полтора-два раза у продуктов этого языка медленнее, чем на С.

Тем не менее, на данный момент преимущества языка Java перевешивают его недостатки, он остаётся одним из самых востребованных.

С# - «С Sharp» (шарп) язык, представленный корпорацией Microsoft. Это относительно новый, частично основанный на С / С + + язык, который также пытается включить функциональность Java без кросс-платформенности. Этот язык предназначен для опытных программистов и ограничивается только операционными системами Microsoft.

HTML - HyperText Markup Language (Язык разметки гипертекста) - стандартный язык для веб-браузеров для интерпретации информации из интернета. В HTML производится форматирование текста, изображения, цвета, фоны, создания ссылки. С помощью HTML текст, рисунки, анимация, видео- и аудиоклипы размещаются на отдельных Web-страницах, из которых состоит сайт. Язык HTML позволяет помещать на страницы гипертекстовые ссылки и интерактивные кнопки, которые соединяют между собой Web-страницы одного или разных Web-сайтов. HTML – это язык разметки текста, а не язык программирования.

PHP – данный язык расшифровывается как «PHP Препроцессор Гипертекста» [19, 33, 34]. PHP является языком веб-разметки, позволяет использовать переменные, а также позволяет подключаться к базам данных и выводить информацию на веб-страницы, из-за чего появляется возможность создавать более сложный контент, чем HTML.

После проведения анализа и сравнения основных языков программирования я выбрала для себя приоритетным языком PHP.

Технология языка PHP дает возможность превратить веб-тренажер из набора статических Web-страниц в выполняемое на сервере приложение,

интерфейс которого представляет собой Web-страницы. В то время как клиентские языки сценариев (например, JavaScript) выполняются Web-браузером, PHP представляет собой серверный язык, т.е. PHP-сценарий, внедренный на Web-страницу, выполняется Web-сервером (а не Web-браузером).

Язык программирования PHP был выбран по следующим причинам:

- Широкая распространенность языка.

Язык PHP является одним из самых распространенных скриптовых языков программирования для Web и поддерживается большинством хостинг-провайдеров.

- Кроссплатформенность.

Реализации языка PHP существуют под все самые популярные операционные системы (Linux, Unix, MacOS, Windows) и веб-серверы (Apache, Microsoft IIS, nginx и т.д.). Существуют уже готовые пакеты для различных операционных систем, позволяющие с легкостью установить набор необходимого программного обеспечения (веб-сервер, СУБД, платформа для разработки) для создания и поддержки веб-сайтов.

- Свободное программное обеспечение.

PHP является бесплатной и свободно распространяемой платформой с открытым исходным кодом, выпускаемой под собственной лицензией.

- Поддержка технологии ODBC.

PHP поддерживает технологию ODBC (Open Database Connectivity – это программный интерфейс (API) доступа к базам данных), что облегчает работу с базой данных заявленного проекта (MS Access).

Для реализации заданных функций необходимо взаимодействие с базой данных, и именно это возможно на PHP. Что же касается баз данных, то стоит аналогично разобраться в них и выбрать наиболее подходящую.

Базы данных - это хранилище для различных типов данных. Каждая база данных, имеет определённую модель (реляционная, документно-ориентированная), которая обеспечивает удобный доступ к данным. Системы

управления базами данных (СУБД) - специальные приложения (или библиотеки) для управления базами данных различных размеров и форм.

MySQL

MySQL поддерживает хранимые процедуры и функции, обработчики ошибок, курсоры, триггеры, представления (не знаю, как без этого сделать нормальную промышленную систему). В MySQL используется немного «подрезанный» стандартный SQL. MySQL поддерживает очень большой список платформ. Надо сказать, что почти все хостеры предоставляют хостинг вместе с MySQL – поэтому она и является такой популярной.

ORACLE

ORACLE отличается в основном тем, что в ней есть еще ряд особенных функций: объектно-ориентированные свойства, автономные транзакции, последовательности, пакеты, аналитические функции. Для работы можно использовать как стандартный SQL, так и язык программирования PL/SQL – в него добавлено множество дополнительных функций, упрощающих работу программиста. Кроме того, последняя на данный момент версия СУБД ORACLE 11g – это первая в мире база данных, специально разработанная для работы в сетях распределенных вычислений Grid. ORACLE, также, как и MySQL – поддерживает очень много различных платформ.

Таким образом, для реализации веб-тренажера будет использоваться язык PHP и СУБД MySQL.

3.3 Описание основного принципа работы реализуемого веб-тренажера

Для того, чтобы студент смог просматривать лекционный материал, проходить тестирования, общаться с преподавателем, добавлять новый материал, добавлять тестовые задания, иметь собственный профиль, добавлять информационные блоки, ему необходимо авторизоваться в системе.

Рассмотрим работу процедуры авторизации при помощи блок-схемы. Блок-схема – распространенный тип схем, описывающих алгоритмы или

процессы, в которых отдельные шаги изображаются в виде блоков различной формы, соединенных между собой линиями.

Блок-схема процесса «Авторизация» изображена на рисунке 3.3.

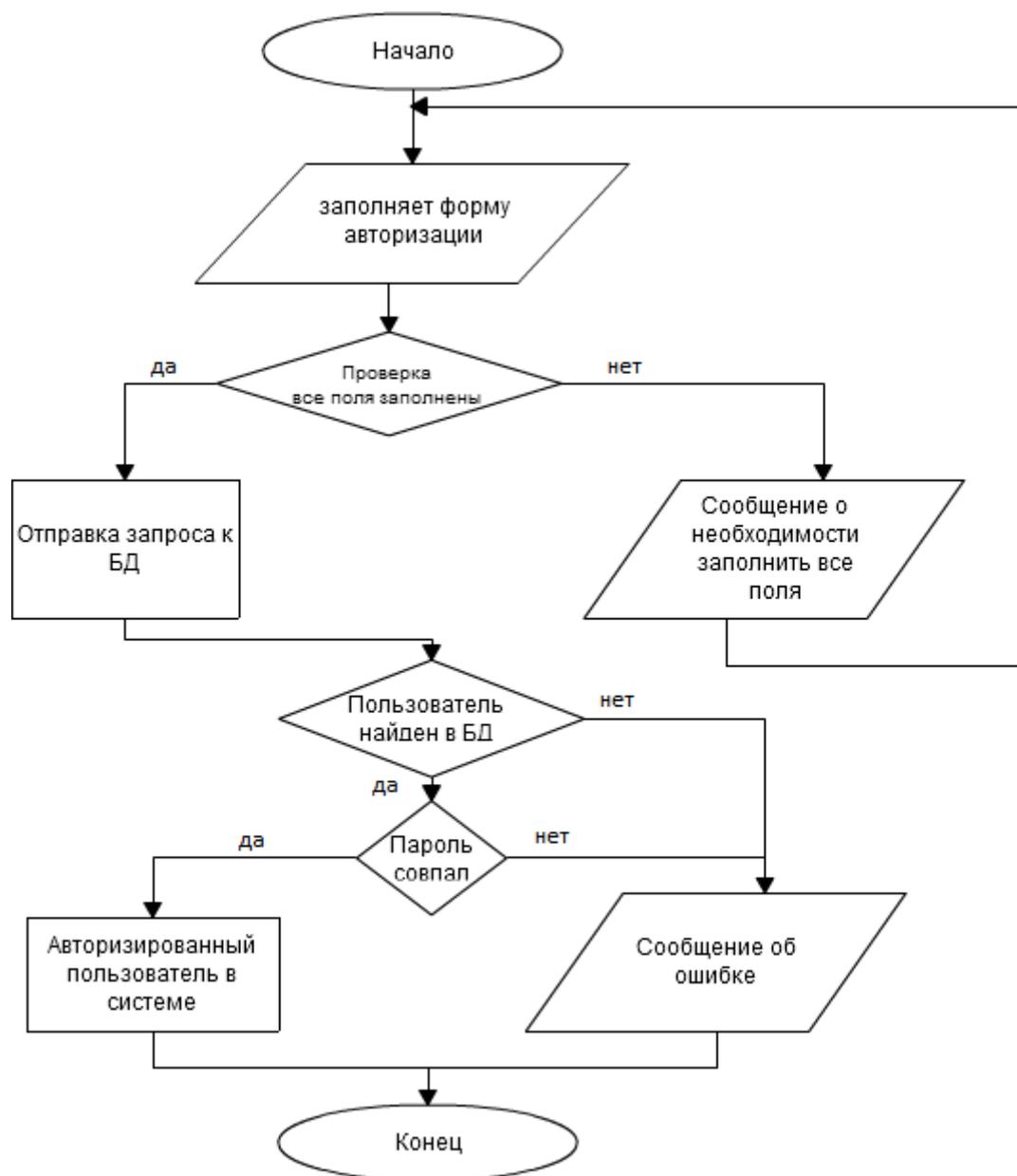


Рисунок 3.3 - Блок-схема процесса «Авторизация»

При работе в режиме преподавателя предусмотрена функция добавления нового материала. Алгоритм добавления информации представлен на рисунке 3.4.

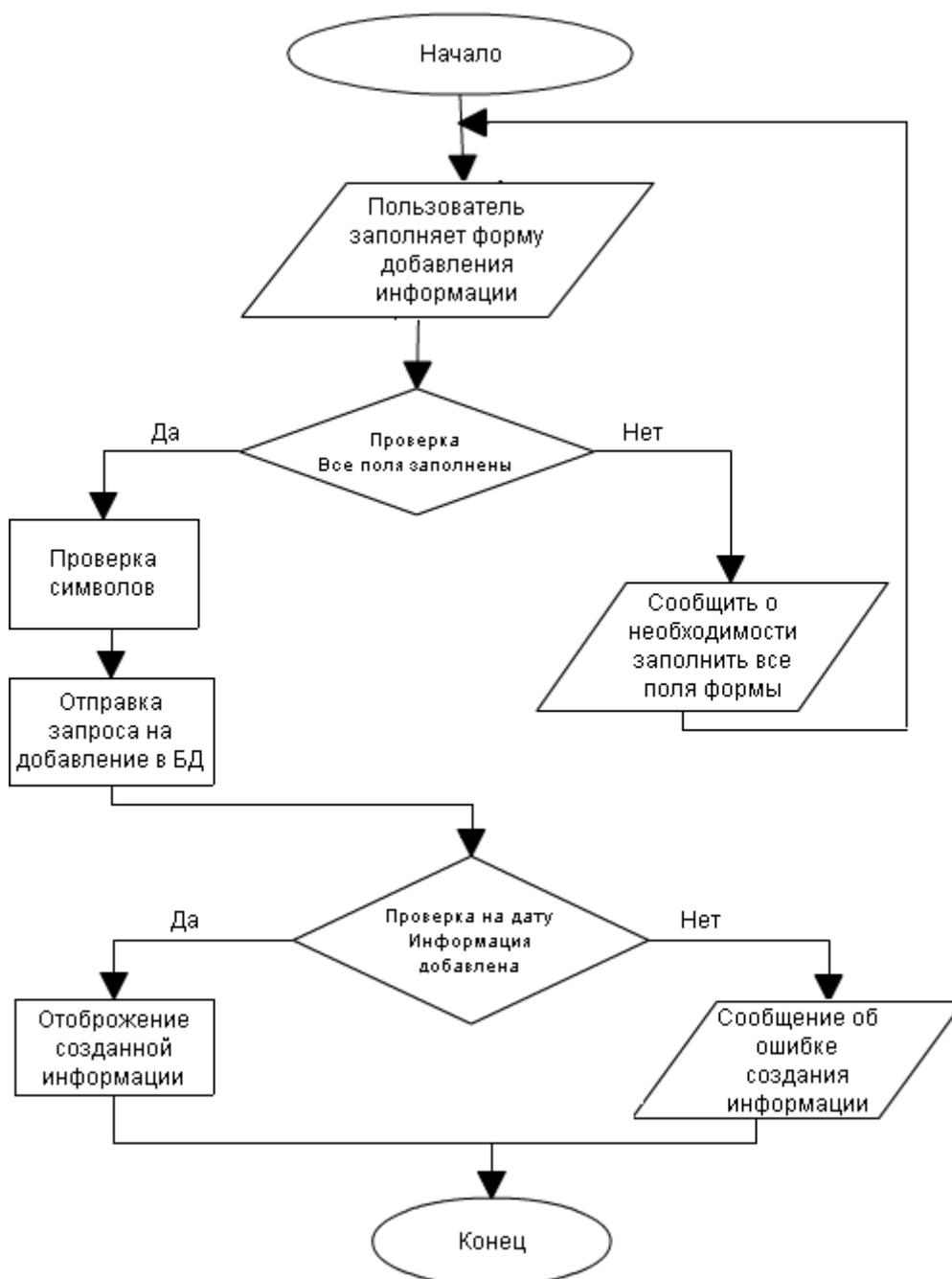


Рисунок 3.4 – Алгоритм добавления информации

Процесс работы веб-тренажера изобразим с помощью блок-схемы (рис. 3.5). В любой момент работы с приложением пользователь может завершить работу, и в данном случае все его действия автоматически отправятся в систему на обработку.

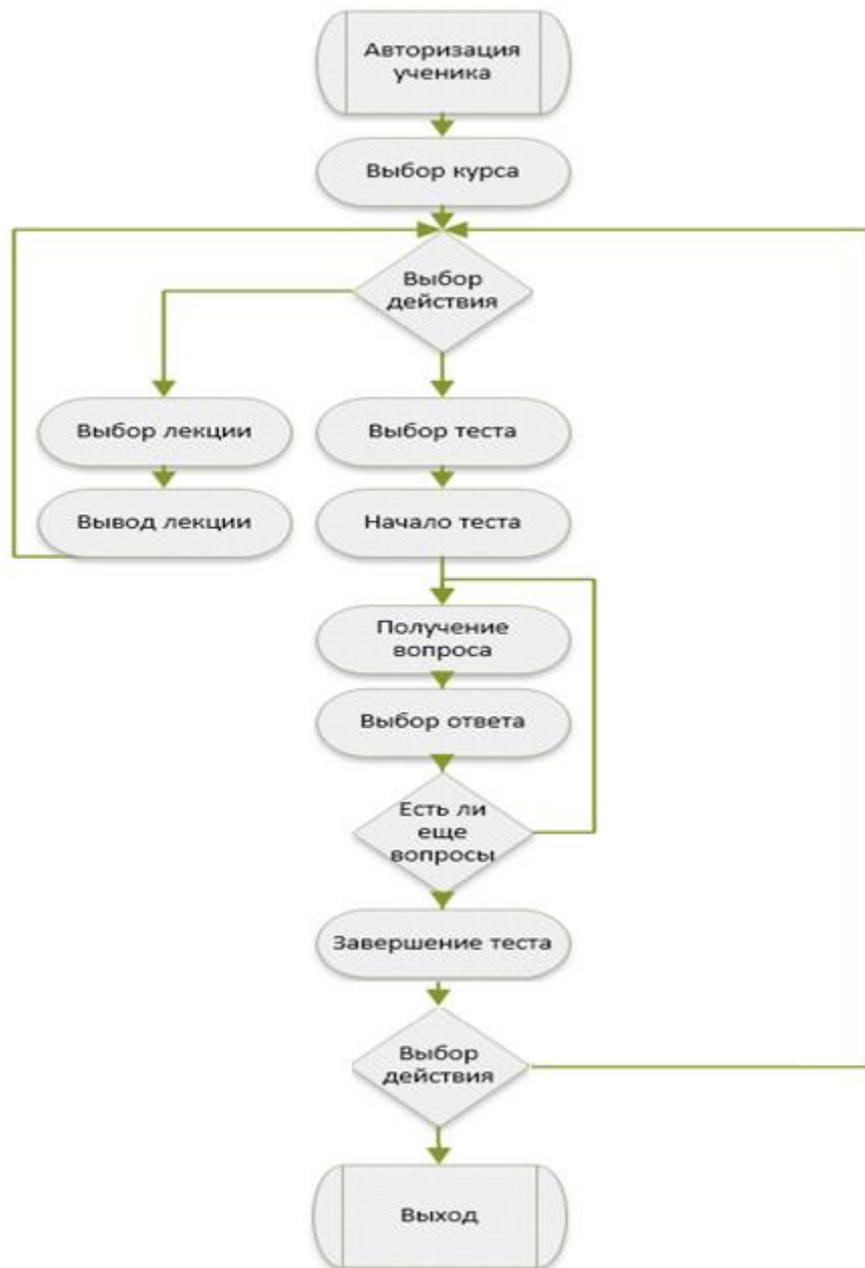


Рисунок 3.5 – Структура представления работы с тестовым материалом

Теоретический материал, представленный в виде лекций в веб-тренажере, формировался в отдельные текстовые файлы, которые подвергались форматированию и редактированию в соответствии с требованиями ГОСТ (рис.3.6).

Также в веб-тренажере по данной дисциплине представлено огромное количество теоретического материала. Вся информация разбита по разделам и каждый студент, пользующийся данным веб-тренажером, может самостоятельно выбрать необходимый ему раздел дисциплины предварительно ознакомившись с его вводной информацией.

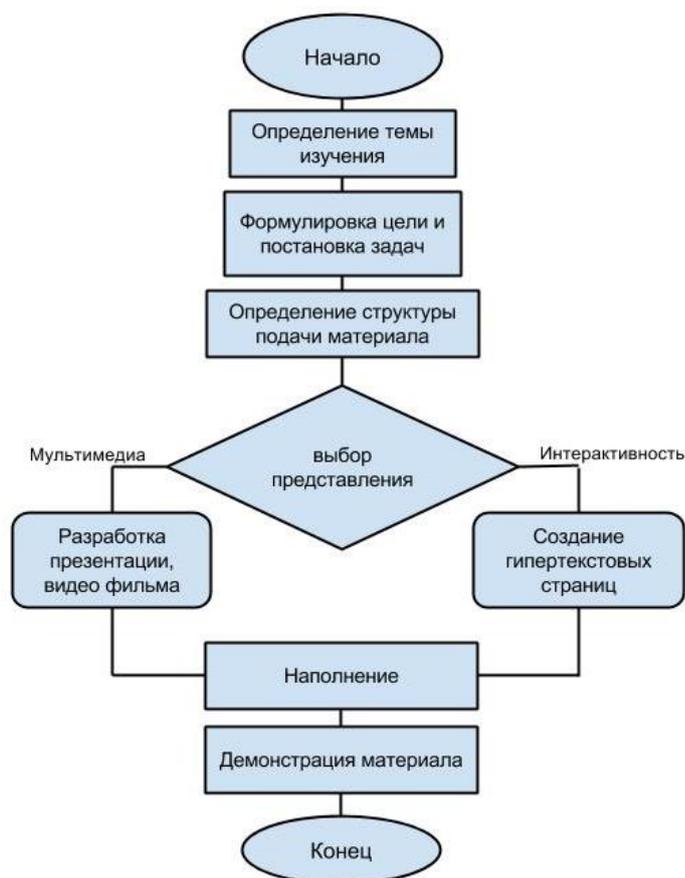


Рисунок 3.6 – Структура представления теоретического материала

На рисунке 3.7 представлена экранная страница, отображающая работу веб-тренажера в режиме просмотра теоретического материала.

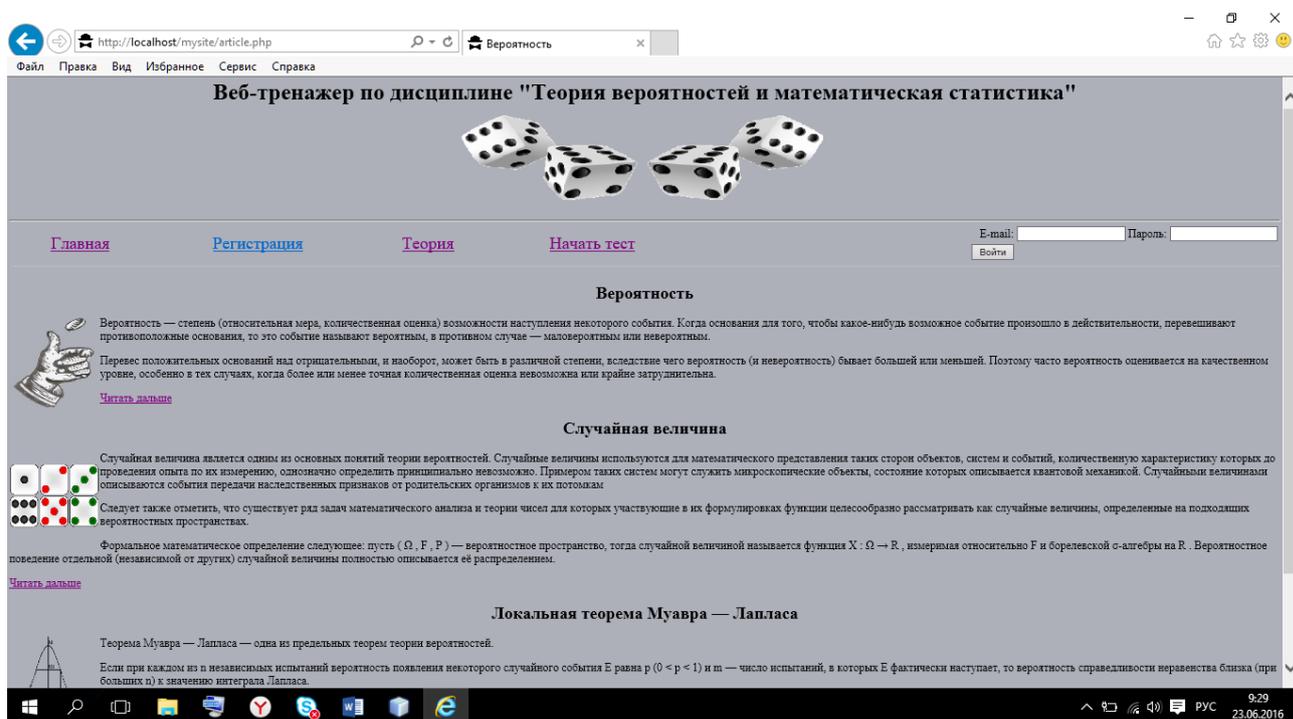


Рисунок 3.7 – Структура представления теоретического материала

В конце каждой подтемы предоставляется проверочное тестирование по пройденному материалу (рис. 3.8).

Вероятность — степень (относительная мера, количественная оценка) возможности наступления некоторого события, когда основания для того, чтобы какое-нибудь возможное событие произошло в действительности, перевешивают противоположные основания, то это событие называют вероятным, в противном случае — маловероятным или невероятным.

Перевес положительных оснований над отрицательными, и наоборот, может быть в различной степени, вследствие чего вероятность (и невероятность) бывает большей или меньшей. Поэтому часто вероятность оценивается на качественном уровне, особенно в тех случаях, когда более или менее точная количественная оценка невозможна или крайне затруднительна.

Определения вероятности Классическое «определение» вероятности исходит из понятия равновероятности как объективного свойства изучаемых явлений. Равновероятность является неопределяемым понятием и устанавливается из общих соображений симметрии изучаемых явлений. Например, при подбрасывании монетки исходят из того, что в силу предполагаемой симметрии монетки, однородности материала и случайности (непредвзятости) подбрасывания нет никаких оснований для предпочтения «решки» перед «орлом» или наоборот, то есть выпадение этих сторон можно считать равновероятными (равновероятными). Наряду с понятием равновероятности в общем случае для классического определения необходимо также понятие элементарного события (исхода), благоприятствующего или нет изучаемому событию A . Речь идет об исходах, наступление которых исключает возможность наступления иных исходов. Это несовместимые элементарные события. К примеру при бросании игральной кости выпадение конкретного числа исключает выпадение остальных чисел. Классическое определение вероятности можно сформулировать следующим образом. Вероятностью случайного события A называется отношение числа n несовместимых равновероятных элементарных событий, составляющих событие A , к числу всех возможных элементарных событий N : $P(A) = n/N$. Например, пусть подбрасываются две кости. Общее количество равновероятных исходов (элементарных событий) равно 36 (так как на каждой из 6 возможных исходов одной кости возможно по 6 вариантов исхода другой). Оценим вероятность выпадения семи очков. Получить 7 очков можно лишь при следующих сочетаниях исходов броска двух костей: 1+6, 2+5, 3+4, 4+3, 5+2, 6+1. То есть всего 6 равновероятных исходов, благоприятствующих получению 7 очков, из 36 возможных исходов броска костей. Следовательно, вероятность будет равна $6/36$ или, если сократить, $1/6$. Для сравнения: вероятность получения 12 очков или 2 очков равна всего $1/36$ — в 6 раз меньше.

Геометрическое определение. Несмотря на то, что классическое определение является интуитивно понятным и выведенным из практики, оно, как минимум, не может быть непосредственно применено в случае, если количество равновероятных исходов бесконечно. Ярким примером бесконечного числа возможных исходов является ограниченная геометрическая область G , например, на плоскости, с площадью S . Случайно «подброшенная» «точка» с равной вероятностью может оказаться в любой точке этой области. Задача заключается в определении вероятности попадания точки в некоторую подобласть g с площадью s . В таком случае, обобщая классическое определение, можно прийти к геометрическому определению вероятности попадания в подобласть g : $P(A) = s/S$. В виду равновероятности вероятность эта не зависит от формы области g , она зависит только от ее площади. Данное определение естественно можно обобщить и на пространство любой размерности, где вместо площади использовать понятие «объема». Более того, именно такое определение приводит к современному аксиоматическому определению вероятности. Понятие объема обобщается до понятия меры некоторого абстрактного множества, к которой предъявляются требования, которыми обладает и «объем» в геометрической интерпретации — в первую очередь, это неотрицательность и аддитивность.

$$P(A) = \lim_{N \rightarrow \infty} \frac{n}{N}$$

Классическое определение при рассмотрении сложных проблем наталкивается на трудности непреодолимого характера. В частности, в некоторых случаях выявить равновероятные случаи может быть невозможно. Даже в случае с монеткой, как известно, существует явно не равновероятная возможность выпадения «ребра», которую из теоретических соображений оценить невозможно (можно только сказать, что оно маловероятно и то это соображение скорее практическое). Поэтому еще на заре становления теории вероятностей было предложено альтернативное «частотное» определение вероятности. А именно, формально вероятность можно определить как предел частоты наблюдений события A , предполагая однородность наблюдений (то есть одинаковость всех условий наблюдений) и их независимость друг от друга, где N — количество наблюдений, а n — количество наступлений события A . Несмотря на то, что данное определение скорее указывает на способ оценки неизвестной вероятности — путем большого количества однородных и независимых наблюдений — тем не менее в таком определении отражено содержание понятия вероятности. А именно, если событие приписывается некоторая вероятность, как объективная мера его возможности, то это означает, что при фиксированных условиях и многократном повторении мы должны получить частоту его появления, близкую к p (тем более близкую, чем больше наблюдений). Собственно, в этом заключается исходный смысл понятия вероятности. В основе лежит объективистский взгляд на явления природы. Ниже будут рассмотрены так называемые законы больших чисел, которые дают теоретическую основу (в рамках излагаемого ниже современного аксиоматического подхода) в том числе для частотной оценки вероятности.

[Начать тестирование по главе](#)

Рисунок 3.8 – Структура подачи теоретического материала с ссылкой на тест

Используемые представления теоретического материала позволяют отобразить основные этапы работы над созданием макета веб-тренажера и демонстрацию взаимодействия электронного ресурса.

На рисунке 3.9 представлен алгоритм работы по созданию электронного практикума, который является одним из основных компонентов веб-тренажера. Практикум предназначен для закрепления полученных знаний путем неоднократного решения подобных задач по изучаемой теме.

Работа над созданием электронного практикума веб-тренажера по дисциплине предполагает выбор действий по созданию интерактивных компонентов: учебных видео фильмов по выполнению практического задания или самостоятельное решение с возможностью проверки полученного результата с использованием прилагаемого решения.

Представленная на рисунке 3.9 схема позволяет отобразить основные этапы работы над созданием макета веб-тренажера, связанного с представлением практического материала по дисциплине.

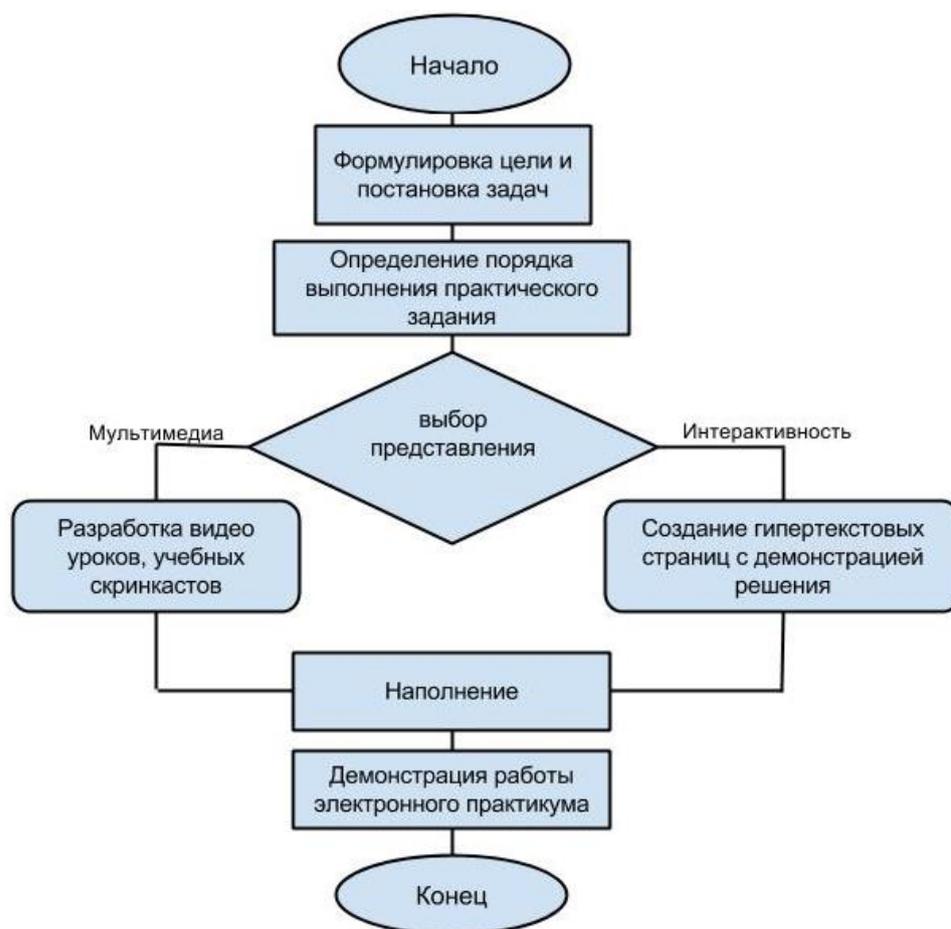


Рисунок 3.9 - Структура представления практического материала

Контроль знаний происходит при помощи тестов. Тест состоит из отдельных заданий тестовой формы, которые располагаются по возрастающему уровню сложности и снабжены инструкцией к выполнению. Все задания теста относятся к одному предметному содержанию.

Добавление нового теста в систему предоставляется только пользователям категории «Преподаватель». Все тесты хранятся в базе данных в таблице Tests. Все тесты относятся к какой-либо категории (таблица Categories), отображение тестов в меню организовано через категорию. В файле шаблона системы происходит вызов метода «module» основного класса «Website», в котором подключается один из модулей.

Отрывок из кода файла шаблона:

```

...
<div class="sidebar1">
  <div class="sidebar1_header"></div>
  <div class="sidebar1_main">

```

```

<?php $this->module("menu", "test", "category_tests", "", "nav"); ?>
</div>
<div class="sidebar1_footer"></div>
</div>
...

```

В данном случае подключается модуль с именем menu, в него передается параметр «test», что означает что мы будем работать с тестами, параметр «category_tests» указывает тесты какой категории будут извлекаться из базы данных, а также 2 необязательных параметра, которые необходимы для оформления.

Процесс выполнения тестовых материалов завершается сообщением с результатами тестирования и возвращением на главный экран.

Реализация тестовой системы как одного из основных компонентов веб-тренажера позволяет организовать проверку знаний.

Таким образом, были рассмотрены принципы организации работы с теоретическим и тестовым материалом, что позволило определить общую структуру представления будущего веб-тренажера. И соответственно необходимо рассмотреть принципы построения его интерфейса для взаимодействия с пользователями.

Общий интерфейс веб-страниц оболочки показан на рисунке 3.10.

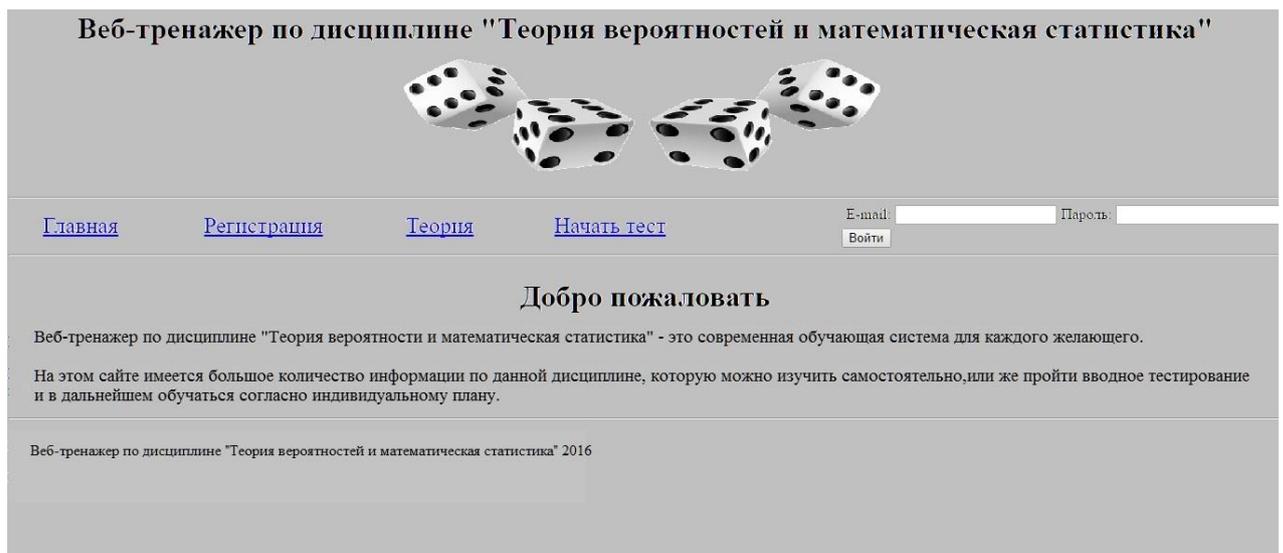


Рисунок 3.10 – Общий вид главной страницы

На рисунке 3.11 представлен вид страницы входного тестирования, обеспечивающего подбор индивидуального маршрута обучения.

Бросается 5 монет. Вероятность того, что выпадет 3 герба, равна:

5/10 4/130 5/16 5/12

15% всех мужчин и 5% всех женщин — дальтоники. Наугад выбранное лицо оказалось дальтоником. Вероятность того, что это мужчина, равна (число мужчин и женщин считается одинаковым) ...

12 1 0,75 0.1

20% всех мужчин и 5% всех женщин — дальтоники. Наугад выбранное лицо оказалось дальтоником. Вероятность того, что это мужчина, равна (число мужчин и женщин считается одинаковым) ...

19 1 0,8 9

$DX = 1,5$. Используя свойства дисперсии, найдите $D(2X + 5)$:

4 5 6 3

$MX = 1,5$. Используя свойства математического ожидания, найдите $M(2X+5)$

1 2 8 10

$MX = 5, MY = 2$. Используя свойства математического ожидания, найдите $M(2X - 3Y)$:

3 5 4 0

X и Y — независимы. $DX = 5, DY = 2$. Используя свойства дисперсии, найдите $D(2X + 3Y)$:

30 39 38 10

Акция приобретена на рынке по цене 1100 руб., дивидент составил 100 руб., показатель рендинга равен:

9,0% 59,09% 9,09% 04%

Рисунок 3.11 – Общий вид входного тестирования

При входном тестировании студент отвечает на 10 вопросов по всем разделам дисциплины «Теория вероятностей и математическая статистика». При завершении входного тестирования незарегистрированный студент видит на экране результат тестирования с количеством правильных и неправильных ответов, также на экране предоставлена ссылка «Вернуться в главное меню» (рисунок 3.12).

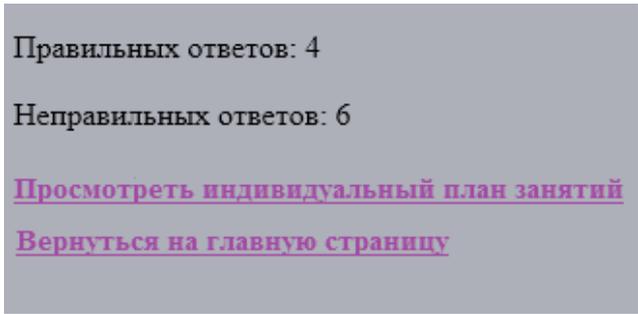
Правильных ответов: 4

Неправильных ответов: 6

[Вернуться на главную страницу](#)

Рисунок 3.12 – Результат входного тестирования

По завершению входного тестирования зарегистрированный студент также видит результаты, но помимо ссылки «Вернуться на главную страницу», появляется ссылка «Просмотреть индивидуальный план занятий» (рисунок 3.11), по которой в дальнейшем и пойдет его персональное обучение.



Правильных ответов: 4
Неправильных ответов: 6
[Просмотреть индивидуальный план занятий](#)
[Вернуться на главную страницу](#)

Рисунок 3.13 – Результат входного тестирования для зарегистрированного студента

Таким образом, был реализован веб-тренажер, представленный в виде обучающей системы. Данный обучающий ресурс предназначен для интерактивного взаимодействия со студентами и позволяет не только предоставлять изучаемый материал, но и проверять уровень усвоения полученных знаний.

Вывод по главе

Представленный в третьей главе веб-тренажер по дисциплине «Теория вероятностей и математическая статистика» является удобным источником информации и системой для проверки полученных знаний одновременно.

Заключение

В данной бакалаврской работе была поставлена цель – разработка веб-тренажера по дисциплине "Теория вероятностей и математическая статистика".

В ходе выполнения бакалаврской работы была проанализирована необходимая литература, с помощью которой были определены основные требования, предъявляемые к разработке веб-тренажеров по учебной дисциплине.

Для выполнения поставленной цели в бакалаврской работе был проведен анализ предметной области, рассмотрены основные требования к системе проектирования веб-тренажера, методические рекомендации по разработке веб-тренажера, требования, предъявляемые к дизайну, структурная организация. Также был проведён сравнительный анализ веб-тренажеров.

Была подробно исследована предметная область, на основе анализа которой были определены требования к функциональным характеристикам разрабатываемого веб-тренажера по дисциплине «Теория вероятностей и математическая статистика».

Для моделирования компонентов веб-тренажера были построены логическая и физическая модели данных, которые позволяют более полно оценить специфику создаваемого программного продукта.

Выбран комплекс технических и программных средств реализации. Система реализована в трехзвенной архитектуре «клиент-сервер» на языке PHP. В качестве системы базы данных использована СУБД MySQL. К достоинствам разработки можно отнести простоту в эксплуатации и сопровождения, низкую стоимость владения.

Задачи, выполненные в ходе работы, позволили в итоге создать веб-тренажер с элементами динамического сайта, обеспечивающий качественную подготовку студентов по дисциплине.

Список используемой литературы

Учебники и учебные пособия

1. Александров, Д.В. Инструментальные средства информационного менеджмента. CASE-технологии и распределенные информационные системы: Учебное пособие / Д.В. Александров. - М.: ФиС, 2011. - 224 с.
2. Алистер Коберн. Современные методы описания функциональных требований к системам. – М: Лори, 2011. 288 с.
3. Ахметсафина Р.З., Карданова Е.Ю., Серова А.В., Чернова Л.М. Спецификация компетентностно-ориентированных оценочных средств // В кн.: Проблемы качества образования. Материалы XXIII Всероссийской научно-методической конференции 20-27 мая 2013 г. М., Уфа: УГАТУ. 2013.
4. Баженов Р. И., Корнилков А. П., Лопатин Д. К. Проектирование web-ориентированной информационной системы университета на основе клиент-серверных технологий // Актуальные проблемы гуманитарных и естественных наук. 2014. №4. С. 68-71.
5. Баженов Р. И., Лопатин Д. К. О применении современных технологий в разработке интеллектуальных систем // Журнал научных публикаций аспирантов и докторантов. 2014. № 3 (93). С. 263-264.
6. Большаков А.А., Вешнева И.В., Мельников Л.А., Перова Л.Г. Применение математического аппарата теории нечетких множеств к задачам управления вузом на основе сбалансированной системы показателей учреждения // Системы управления и информационные технологии. 2011. №1.1(43).
7. Брукс П. Проектирование процесса проектирования. Монография. М. ООО И.Д. Вильямс, 2013. - 464 стр.
8. Гаврилов С. И. Модели, методы и программные средства оценки качества информационно- образовательных ресурсов : автореф. дис.... канд. технич. Наук. М., 2011.

9. Гудвин, Г.К. Проектирование систем управления / Г.К. Гудвин, С.Ф. Гребен, М.Э. Сальгадо; Пер. с англ. А.М. Епанешников. - М.: БИНОМ. ЛЗ, 2012. - 911 с.
10. Ежик, И.Г. Научно-методическое обеспечение учебного информационного взаимодействия в группе на базе интерактивных электронных образовательных ресурсов (на примере обучения курсантов английскому языку в военном вузе): автореф. канд. ... пед. наук: 13.00.02 / Ежик Ирина Григорьевна. – М, 2013, – 18 с.
11. Заботина, Н.Н. Проектирование информационных систем / Н.Н. Заботина. – М.: Инфра-М, 2013. – 336 с.
12. Затонский, А.В. Информационные технологии. Разработка информационных моделей и систем. Учебное пособие / А.В. Затонский. – М.: РИОР, Инфра-М, 2014. 344 с.
13. Калинина, В.Н. Теория вероятностей и математическая статистика: Учебник для бакалавров / В.Н. Калинина. - М.: Юрайт, 2013. - 472 с.
14. Каменева Т. Н. Педагогические технологии в электронном образовательном пространстве: традиции и инновации //Образовательные технологии и общество. – 2013. – Т. 16. – №. 1. – С. 609-626.
15. Кирилова Г. И. Развитие и саморазвитие информационной образовательной среды профессионального образования //Образовательные технологии и общество (Educational Technology & Society). – 2012. – Т. 15. – №. 3. – С. 358-368.
16. Киселев Г. М. Теория и практика информатизации профессионального образования в вузах : Моно- графия. М. : МРСЭИ, 2013.
17. Климов, Г.П. Теория вероятностей и математическая статистика / Г.П. Климов. - М.: МГУ, 2011. - 368 с.
18. Колемаев, В.А. Теория вероятностей и математическая статистика: Учебник / В.А. Колемаев, В.Н. Калинина. - М.: КноРус, 2013. - 376 с.
19. Колисниченко Д. Php и MySQL. Разработка Web-приложений / Д. Колисниченко – Спб.: БХВ-Петербург, 2015. 592 с.

20. Кочетков, Е.С. Теория вероятностей и математическая статистика: Учебник / Е.С. Кочетков, С.О. Смерчинская, В.В. Соколов. - М.: Форум, НИЦ ИНФРА-М, 2013. - 240 с.

21. Кремер, Н.Ш. Теория вероятностей и математическая статистика: Учебник для студентов вузов / Н.Ш. Кремер. - М.: ЮНИТИ-ДАНА, 2012. - 551 с.

22. Кушников В.А., Яндыбаева Н.В. Оценка качества образовательного процесса в вузе на основе модели Форрестера // Вестник Саратовского государственного технического университета. 2011.

23. Левкин Г. Г., Глухих В. Р., Базилевич С. В. Организация дистанционного обучения в профессиональной переподготовке: Аспекты использования социальных сетей, электронных учебников и учебного сайта в дополнительном образовании. – Scientific magazine» Kontsep, 2012.

24. Насс, О.В. Теоретико-методические основания формирования компетентности преподавателей в области создания электронных образовательных ресурсов (на базе адаптивных инструментальных комплексов): автореф. дис. ... д-ра пед. наук: 13.00.02 / Насс Оксана Викторовна. – М., 2013. – 43 с.

25. Первезенцева, Э.А. Разработка комплекса электронных образовательных ресурсов и его использование для самостоятельной информационной учебной деятельности: дис. ... канд. пед. наук: 13.00.02 / Первезенцева Эвелина Александровна. – М., 2013, – 189 с.

26. Татаринцев, А. И. Электронный учебно-методический комплекс как компонент информационно-образовательной среды педагогического вуза [Текст] / А. И. Татаринцев // Теория и практика образования в современном мире: материалы междунар. науч. конф. (г. Санкт-Петербург, февраль 2012 г.). — СПб.: Реноме, 2012. — С. 367-370.

27. Федорова, Г.Н. Информационные системы : Учебник. / Г.Н. Федорова. – М.: Академия, 2013. – 208 с.

28. Шалкина, Т.Н. Электронные учебно-методические комплексы: проектирование, дизайн, инструментальные средства / Т.Н. Шалкина, В.В. Запорожко, А.А. Рычкова. – Оренбург, ГОУ ОГУ, 2008. – 160 с.

29. Щеголева Л.В., Кириленко А.Н. Проектирование информационной системы: структурный подход. - Петрозаводск: Издательство ПетрГУ, 2013. - 104 с.

Литература на иностранном языке

30. Andre C. SyncCharts: A Visual Representation of Reactive Behaviors /Tech. Rep. RR 95-52,13 S, Sophia-Antipolis, France, 1995.

31. Encontre V. SDL: a standard language for Ada real-time applications /Proceedings of the conference on TRI-Ada '91, pp.45-53, October 21-25, 1991, San Jose, California, United States.

32. Maraninchi F. The Argos language: Graphical representation of automata and description of reactive systems /Presented at the IEEE Workshop Visual Lang., Kobe, Japan, 1991.

33. Ollson, M. PHP 7 Quick Scripting Reference/ 2th edition, - Apress. 2016.

34. Sklar, D. Learning PHP: A Gentle Introduction to the Web's Most Popular Language/ - O'Reilly Media. 2016.

Главная страница веб-тренажера

```

<!DOCTYPE html>
<html>
<head>
  <title>Веб-тренажер по дисциплине "Теория вероятностей и математическая статистика"</title>
  <meta http-equiv="Content-type" content="text/html; charset=utf-8" />
  <link rel="stylesheet" type="text/css" href="styles/main.css" />
</head>
<body>
  <table id="main">
    <?php
      require_once "blocks/top.php";
    ?>
    <tr>
      <td colspan="2">
        <hr />
      </td>
    </tr>
    <tr>
      <td colspan="2">
        <table cellpadding="0" cellspacing="0" id="content">
          <tr>
            <td>
              <div class="article">
                <h1>Добро пожаловать</h1>
                <p>Веб-тренажер по дисциплине
"Теория вероятности и математическая статистика" - это современная обучающая система для каждого
желающего. </p>
                <p>На этом сайте имеется большое
количество информации по данной дисциплине, которую можно изучить самостоятельно,или же пройти
вводное тестирование и в дальнейшем обучаться согласно индивидуальному плану.</p>
              </div>
            </td>
          </tr>
        </table>
      </td>
    </tr>
    <tr>
      <td colspan="2">
        <hr />
      </td>
    </tr>
  </table>

```

```

        </td>
    </tr>
<?php
    require_once "blocks/footer.php";
?>
</table>
</body>
</html>

```

Страница с кратким описанием каждой статьи

```

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <style>
    body {
        background: #ADB0B9; /* Цвет фона */
        color: #000000; /* Цвет текста */
    }
    </style>
    <title>Теория</title>
    <meta http-equiv="Content-type" content="text/html; charset=utf-8" />
    <link rel="stylesheet" type="text/css" href="styles/main.css" />
</head>
<body>
    <table id="main">
        <?php
            require_once "blocks/top.php";
        ?>
        <tr>
            <td colspan="2">
                <hr />
            </td>
        </tr>
        <tr>
            <td colspan="2">
                <table cellpadding="0" cellspacing="0" id="content">
                    <title> текст обтекает картинку справа</title>
                    <tr>
                        <td>
                            <?php
                                require_once "blocks/full_article.php";
                            ?>
                        </td>

```

```

</tr>
</head>
<body>
  <div class="article">
    <h2>Вероятность</h2>
    <p class="article_img">
      

```

<p>Вероятность — степень (относительная мера, количественная оценка) возможности наступления некоторого события. Когда основания для того, чтобы какое-нибудь возможное событие произошло в действительности, перевешивают противоположные основания, то это событие называют вероятным, в противном случае — маловероятным или невероятным.</p>

<p>Перевес положительных оснований над отрицательными, и наоборот, может быть в различной степени, вследствие чего вероятность (и невероятность) бывает большей или меньшей. Поэтому часто вероятность оценивается на качественном уровне, особенно в тех случаях, когда более или менее точная количественная оценка невозможна или крайне затруднительна.</p>

```

<p>
  <a href="article.php?id=1">Читать дальше</a>
</p>
</div>

```

```

<div class="article">
  <h2>Случайная величина</h2>
  <p class="article_img">
    

```

<p>Случайная величина является одним из основных понятий теории вероятностей. Случайные величины используются для математического представления таких сторон объектов, систем и событий, количественную характеристику которых до проведения опыта по их измерению, однозначно определить принципиально невозможно. Примером таких систем могут служить микроскопические объекты, состояние которых описывается квантовой механикой. Случайными величинами описываются события передачи наследственных признаков от родительских организмов к их потомкам</p>

<p> Следует также отметить, что существует ряд задач математического анализа и теории чисел для которых участвующие в их формулировках функции целесообразно рассматривать как случайные величины, определенные на подходящих вероятностных пространствах.</p>

<p> Формальное математическое определение следующее: пусть (Ω, \mathcal{F}, P) — вероятностное пространство, тогда случайной величиной называется функция $X : \Omega \rightarrow \mathbb{R}$, измеримая относительно \mathcal{F} и борелевской σ -алгебры на \mathbb{R} . Вероятностное поведение отдельной (независимой от других) случайной величины полностью описывается её распределением.</p>

```

<p>
  <a href="article.php?id=1">Читать дальше</a>
</p>
</div>

```

```
<div class="article">
```

```
  <h2>Локальная теорема Муавра — Лапласа</h2>
```

```
  <p class="article_img">
```

```
    
```

```
  <p>Теорема Муавра — Лапласа — одна из предельных теорем теории вероятностей.</p>
```

```
  <p>Если при каждом из  $n$  независимых испытаний вероятность появления некоторого случайного события  $E$  равна  $p$  ( $0 < p < 1$ ) и  $m$  — число испытаний, в которых  $E$  фактически наступает, то вероятность справедливости неравенства близка (при больших  $n$ ) к значению интеграла Лапласа.</p>
```

```
</p>
```

```
  </p>
```

```
<p>
```

```
  <a href="article.php?id=1">Читать дальше</a>
```

```
</p>
```

```
  </p>
```

```
</div>
```

```
</body>
```

```
  </table>
```

```
  </td>
```

```
</tr>
```

```
<tr>
```

```
  <td colspan="2">
```

```
    <hr />
```

```
  </td>
```

```
</tr>
```

```
<?php
```

```
  require_once "blocks/footer.php";
```

```
?>
```

```
</table>
```

```
</body>
```

```
</html>
```

Страница с входным тестированием

```
<form action="result.php" method="POST">
```

```
<html>
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<style>
```

```
body {
```

```
  background: #ADB0B9; /* –вет фона */
```

```
  color: #000000; /* –вет текста */
```

```
}
```

```
</style>
```

```
</head>
```

```

<body>
<div>
  <p>Событий какого вида из перечисленных не существует с точки зрения теории вероятностей?</p>
  <label><input name="q1" type="radio" value="c">Достоверные события</label>
  <label><input name="q1" type="radio" value="b">Невозможные события</label>
  <label><input name="q1" type="radio" value="a">Решающие события</label>
  <label><input name="q1" type="radio" value="d">Случайные события</label>
</div>
<div>
  <p>Балансовая стоимость основных фондов — это стоимость ...</p>
  <label><input name="q2" type="radio" value="c">достигается баланс между производством и
  потреблением, предложением и спросом</label>
  <label><input name="q2" type="radio" value="b">Невозможные события</label>
  <label><input name="q2" type="radio" value="a">по которой они учтены в балансе
  предприятия</label>
  <label><input name="q2" type="radio" value="d">ежегодно
</label>
</div>
<div>
  <p>Бросается 5 монет. Вероятность того, что выпадет 3 герба, равна:
</p>
  <label><input name="q3" type="radio" value="c">5/10</label>
  <label><input name="q3" type="radio" value="b">4/130</label>
  <label><input name="q3" type="radio" value="a">5/16
</label>
  <label><input name="q3" type="radio" value="d">5/12</label>
</div>
<div>
  <p> 15% всех мужчин и 5% всех женщин — дальтоники. Наугад выбранное лицо оказалось
  дальтоником.
  Вероятность того, что это мужчина, равна (число мужчин и женщин считается одинаковым) ...
</p>
  <label><input name="q4" type="radio" value="c">12</label>
  <label><input name="q4" type="radio" value="b">1</label>
  <label><input name="q4" type="radio" value="a">0,75
</label>
  <label><input name="q4" type="radio" value="d">0.1</label>
</div>
<div>
  <p> 20% всех мужчин и 5% всех женщин — дальтоники. Наугад выбранное лицо оказалось
  дальтоником.
  Вероятность того, что это мужчина, равна (число мужчин и женщин считается одинаковым) ...

```

```

</p>
  <label><input name="q5" type="radio" value="c">19</label>
  <label><input name="q5" type="radio" value="b">1</label>
  <label><input name="q5" type="radio" value="a">0,8
</label>
  <label><input name="q5" type="radio" value="d">9</label>
</div>
<div>
  <p> DX = 1,5. Используя свойства дисперсии, найдите D(2X + 5):
</p>
  <label><input name="q6" type="radio" value="c">4</label>
  <label><input name="q6" type="radio" value="b">5</label>
  <label><input name="q6" type="radio" value="a">6
</label>
  <label><input name="q6" type="radio" value="d">3</label>
</div>
<div>
  <p> MX = 1,5. Используя свойства математического ожидания, найдите M(2X+5)
</p>
  <label><input name="q7" type="radio" value="c">1</label>
  <label><input name="q7" type="radio" value="b">2</label>
  <label><input name="q7" type="radio" value="a">8
</label>
  <label><input name="q7" type="radio" value="d">10</label>
</div>
<div>
  <p> MX = 5, MY = 2. Используя свойства математического ожидания, найдите M(2X — 3Y):</p>
  <label><input name="q8" type="radio" value="c">3</label>
  <label><input name="q8" type="radio" value="b">5</label>
  <label><input name="q8" type="radio" value="a">4
</label>
  <label><input name="q8" type="radio" value="d">0</label>
</div>
<div>
  <p> X и Y — независимы. DX = 5, DY = 2. Используя свойства дисперсии, найдите D(2X + 3Y):</p>
  <label><input name="q9" type="radio" value="c">30</label>
  <label><input name="q9" type="radio" value="b">39</label>
  <label><input name="q9" type="radio" value="a">38</label>
  <label><input name="q9" type="radio" value="d">10</label>
</div>

```

```

<div>
  <p>Акция приобретена на рынке по цене 1100 руб., дивидент составил 100 руб., показатель рендинга
  равен:</p>
  <label><input name="q10" type="radio" value="c">9,0%</label>
  <label><input name="q10" type="radio" value="b">59,09%</label>
  <label><input name="q10" type="radio" value="a">9,09%</label>
  <label><input name="q10" type="radio" value="d">04%</label>
</div>
<div>
<button type="submit">Результат</button>

</form>
</body>
</html>

```

Страница с подсчетом и выводом результатов тестирования

```

<!DOCTYPE HTML>
<?php
  $ot = 0;
  $not = 0;
  if ($_POST[q1] == a){$ot++;} else {$not++;}
  if ($_POST[q2] == a){$ot++;} else {$not++;}
  if ($_POST[q3] == a){$ot++;} else {$not++;}
  if ($_POST[q4] == a){$ot++;} else {$not++;}
  if ($_POST[q5] == a){$ot++;} else {$not++;}
  if ($_POST[q6] == a){$ot++;} else {$not++;}
  if ($_POST[q7] == a){$ot++;} else {$not++;}
  if ($_POST[q8] == a){$ot++;} else {$not++;}
  if ($_POST[q9] == a){$ot++;} else {$not++;}
  if ($_POST[q10] == a){$ot++;} else {$not++;}
?>
<html>
<head>
  <style>
  body {
    background: #ADB0B9; /* Цвет фона */
    color: #000000; /* Цвет текста */
  }
</style>
<meta charset="utf-8">
<style>
  body {
    background: #ADB0B9; /* Цвет фона */

```

```
color: #000000; /* Цвет текста */
}
</style>
</head>
<body>

  <p>Правильных ответов: <?php echo $ot; ?></p>
  <p>Неправильных ответов: <?php echo $not; ?></p>
  <input type="button" value="Назад" onclick="history.back()">'
  <p><a href="\mysite\index.php" target="_blank">Вернуться на главную страницу</a></p>

</body>
</html>
```