

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение  
высшего образования

«Тольяттинский государственный университет»

Институт математики физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование кафедры)

02.03.03 Математическое обеспечение и администрирование информационных  
систем

(код и наименование направления подготовки, специальности)

Технология программирования

(направленность (профиль)/специализация)

## БАКАЛАВРСКАЯ РАБОТА

на тему **Разработка системы публикации расписания учебных занятий в  
ФГБОУ ВО «ТГУ» с модулем оповещения**

Студент

В.Э. Панченко

(И.О. Фамилия)

\_\_\_\_\_

(личная подпись)

Руководитель

А.Б. Кузьмичев

(И.О. Фамилия)

\_\_\_\_\_

(личная подпись)

Консультанты

Т.В. Маркелова

(И.О. Фамилия)

\_\_\_\_\_

(личная подпись)

**Допустить к защите**

Заведующий кафедрой к.т.н., доцент, А.В. Очеповский

(ученая степень, звание, И.О. Фамилия)

\_\_\_\_\_

(личная подпись)

« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_\_ г.

Тольятти 2019

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

Институт математики, физики и информационных технологий  
(наименование института полностью)

Кафедра «Прикладная математика и информатика»  
(наименование кафедры)

УТВЕРЖДАЮ

Зав. кафедрой «Прикладная математика и информатика»

\_\_\_\_\_ А.В. Очеповский  
(подпись) (И.О. Фамилия)

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

**ЗАДАНИЕ**

**на выполнение выпускной квалификационной работы**

Студент Панченко Владимир Эдуардович

1. Тема Разработка системы публикации расписания учебных занятий в ФГБОУ ВО «ТГУ» с модулем оповещения.

2. Срок сдачи студентом законченной выпускной квалификационной работы 24.06.2019 г.

3. Исходные данные к выпускной квалификационной работе текущий процесс публикации расписания занятий в ФГБОУ ВО «ТГУ» и систем планирования времени.

4. Содержание выпускной квалификационной работы (перечень подлежащих разработке вопросов, разделов) Анализ существующих технологий системы публикации расписания учебных занятий, проектирование, реализация системы публикации расписания учебных занятий и модуля оповещения.

5. Ориентировочный перечень графического и иллюстративного материала блок-схемы выполнения основных процессов программного приложения; графики и диаграммы, поясняющие результат работы приложения.

6. Консультанты по разделам Т.В. Маркелова

7. Дата выдачи задания «15» декабря 2017г.

Руководитель выпускной  
квалификационной работы \_\_\_\_\_  
(подпись)

А.Б. Кузьмичев  
(И.О. Фамилия)

Задание принял к исполнению \_\_\_\_\_  
(подпись)

В.Э. Панченко  
(И.О. Фамилия)

## АННОТАЦИЯ

Объем 51 с., 9 рис., 10 таблиц, 25 источников (в т.ч. 7 источников на английском языке).

Работа состоит из введения, 3 глав, заключения и приложения.

Объектом работы является система публикации учебных занятий ФГБОУ ВО «ТГУ» с модулем оповещений.

Целью бакалаврской работы является разработка системы публикации учебных занятий ФГБОУ ВО «ТГУ» с модулем оповещений.

Актуальность работы заключается в том, что в ФГБОУ ВО «ТГУ» отсутствует автоматизированная система публикации расписания учебных занятий. Модуль оповещения и интеграция сервиса планирования событий позволит удобнее планировать своё время и совмещать его с расписанием занятий.

Система публикации расписания учебных занятий должна будет получать данные с web-сервера ТГУ, обрабатывать их, и выводить пользователю в виде html-таблицы. Модуль оповещения должен, получать данные с стороннего сервиса планирования событий, например такого как Google-календарь, и отображать их в расписании пользователя. Платформа реализации системы будет NetBeans IDE. Проведя анализ языков программирования, интегрированной среды разработки и web-сервера приложений, был сделан вывод, что наилучшим выбором будет язык Java, интегрированной среды разработки Netbeans IDE, а в качестве web-сервера приложений будет использоваться Wildfly.

В заключении делается вывод по разработке системы публикации расписания учебных занятий и модуля оповещения.

## **ABSTRACT**

Volume 51 p., 7 pic., 8 tables, 25 sources (including 7 sources in English).

The work consists of an introduction, three chapters, a conclusion and an appendix.

The object of the work is the publication system of the training sessions of the FGBOU VO "TSU" with a notification module.

The aim of the bachelor's work is the development of the publication system for the training sessions of the FGBOU VO "TSU" with a notification module.

The urgency of the work lies in the fact that there is no automated system for publishing the schedule of training sessions in the TSBO "TSU". The notification module and the integration of the organizer will make it easier to plan your time, combine it with the schedule of classes.

The system of publication of the schedule of training sessions will have to receive data from the TSU web server, process them, and output it to the user as an html table. The notification module should receive data from a third-party event planning service, such as a Google calendar, and display them in the user's schedule. The implementation platform of the system will be the NetBeans IDE. After analyzing the programming languages, integrated development environment, and the application web server, it was concluded that the best choice would be the Java language, the Netbeans IDE integrated development environment, and Wildfly as the application web-server.

In conclusion, a conclusion is drawn on the development of a publication system for scheduling training sessions and an alert module.

## Содержание

ВВЕДЕНИЕ.....	3
1 АНАЛИЗ СУЩЕСТВУЮЩИХ ТЕХНОЛОГИЙ СОЗДАНИЯ СИСТЕМЫ ВЫВОДА ДАННЫХ СИСТЕМЫ ПУБЛИКАЦИИ РАСПИСАНИЯ УЧЕБНЫХ ЗАНЯТИЙ.....	3
1.1 Анализ предметной области с точки зрения разработки системы публикации расписания учебных занятий на примере ФГБОУ ВО «ТГУ». .3	
1.2 Анализ систем публикации расписания учебных занятий.....	3
1.3 Анализ сервисов планирования событий.....	3
1.4 Выработка требований к будущей системе публикации расписания занятий в ФГБОУ ВО «ТГУ» и модулю оповещения.....	3
1.5 Постановка задачи на разработку системы публикации расписания занятий с модулем оповещения.....	3
2 ПРОЕКТИРОВАНИЕ СИСТЕМЫ ПУБЛИКАЦИИ РАСПИСАНИЯ И МОДУЛЯ ОПОВЕЩЕНИЯ.....	3
2.1 Разработка новой технологии системы публикации расписания занятий с модулем оповещения.....	3
2.2 Разработка архитектуры системы публикации расписания с модулем оповещения.....	3
2.3 Разработка диаграммы последовательности по прецеденту “просмотр расписания группы”.....	3
2.4 Разработка диаграммы классов системы публикации расписания занятий с модулем оповещения.....	3
2.5 Разработка модуля оповещения.....	3
2.6 Обоснование и разработка математической модели классифицирования событий.....	3
3 РЕАЛИЗАЦИЯ СИСТЕМЫ ПУБЛИКАЦИИ РАСПИСАНИЯ И МОДУЛЯ ОПОВЕЩЕНИЯ.....	3
3.1 Выбор средств реализации системы публикации расписания занятий с модулем оповещения.....	3
3.2 Выбор интегрированной среды разработки и web-сервера приложений	3
3.3 Описание основных принципов работы реализуемой системы публикации расписания занятий с модулем оповещения.....	3
3.4 Разработка алгоритма классификации событий.....	3
ЗАКЛЮЧЕНИЕ.....	3
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ.....	3
Приложение А.....	3

## **ВВЕДЕНИЕ**

С течением времени прогресс и информационные технологии, в частности, стремительно развиваются. Соответственно должны развиваться и корпоративные технологии, чтобы не потерять актуальность и востребованность, особенно информационные системы публикации корпоративных данных, таких, как расписание учебных занятий в ВУЗах.

Актуальность работы заключается в том, что в ФГБОУ ВО «ТГУ» отсутствует автоматизированная система публикации расписания учебных занятий. Модуль оповещения и интеграция сервиса планирования событий, позволит удобнее планировать своё время, совмещать его с расписанием занятий.

Новизна данной работы заключается в том, что ранее в ТГУ не существовало единой системы публикации расписания с интегрированным органайзером и модулем оповещений. Реализуя в этой дипломной работе систему публикаций расписания учебных занятий, она будет спроектирована, разработана и внедрена в ФГБОУ ВО «ТГУ».

Объектом работы является система публикации учебных занятий ФГБОУ ВО «ТГУ» с модулем оповещений

Предметом работы является технология разработки системы публикации учебных занятий для ФГБОУ ВО «ТГУ».

Целью бакалаврской работы является разработка системы публикации учебных занятий ФГБОУ ВО «ТГУ» с модулем оповещений. Для достижения цели необходимо выполнить следующие задачи:

- проанализировать текущую систему публикации расписания учебных занятий ФГБОУ ВО «ТГУ»;
- составить список требований к проектируемой системе для публикации учебных занятий;
- спроектировать модуль для публикации учебных занятий ВУЗа;

- разработать систему публикации учебных занятий ФГБОУ ВО «ТГУ»;
- разработать модуль оповещения о событиях в системе публикации расписания учебных занятий в ФГБОУ ВОУ «ТГУ».

# **1 АНАЛИЗ СУЩЕСТВУЮЩИХ ТЕХНОЛОГИЙ СОЗДАНИЯ СИСТЕМЫ ВЫВОДА ДАННЫХ СИСТЕМЫ ПУБЛИКАЦИИ РАСПИСАНИЯ УЧЕБНЫХ ЗАНЯТИЙ**

## **1.1 Анализ предметной области с точки зрения разработки системы публикации расписания учебных занятий на примере ФГБОУ ВО «ТГУ»**

Тольяттинский государственный университет – высшее учебное заведение, созданное в 2001 году решением Правительства России на базе Тольяттинского политехнического института и Тольяттинского филиала Самарского государственного педагогического университета.

ТГУ – крупнейший и старейший государственный вуз города, осуществляющий подготовку по гуманитарно-педагогическому, техническому и естественнонаучному направлениям, а также один из немногих российских вузов имеющий военную кафедру и учебный военный центр [2].

Главный корпус университета расположен по адресу: 445020, г. Тольятти, ул. Белорусская, 14 (центральный кампус). В сети интернет существует сайт университета, расположенный по адресу: <http://www.tltsu.ru> Тольяттинский государственный университет объединяет институты:

1. Архитектурно-строительный институт.
2. Гуманитарно-педагогический институт.
3. Институт изобразительного и декоративно-прикладного искусства.
4. Институт машиностроения.
5. Институт математики, физики и информационных технологий.
6. Институт права.
7. Институт энергетики и электротехники.
8. Институт химии и инженерной экологии.
9. Институт физической культуры и спорта.

10. Институт финансов, экономики и управления.

11. Институт военного обучения.

В структуру ТГУ входят 47 кафедр и работают более 2000 преподавателей и сотрудников, среди которых 100 докторов и более 450 кандидатов наук. В среднем во ФГБОУ ВО «ТГУ» учится 12 000 студентов.

С точки зрения организации учебного процесса каждому ВУЗу необходимо публиковать расписание учебных занятий, и наиболее эффективно эту работу будет выполнять система публикации расписания, в состав которой войдет модуль оповещения.

С точки зрения студента, быстрый доступ к актуальному расписанию занятий играет крайне важную роль. Благодаря этому, студент всегда знает где, когда и какая пара будет. Так же оно даёт возможность быть всегда в курсе изменений, рассчитывать своё время, и вовремя приходить на пары.

Для преподавателей оно имеет так же колоссальную роль. Имея быстрый доступ к актуальному расписанию занятий, преподаватель может планировать своё время, оценивать нагрузку, знает в какой аудитории и у какой группы у него будет следующая пара, что в свою очередь дает ему больше времени, чтобы подготовиться к предстоящему занятию.

Для других же сотрудников ВУЗа, например методистов, лаборантов, это так же необходимо. Имея единую систему, в которой указана информация о паре, какой преподаватель и аудитория, проще составлять расписание, подготавливать аудитории к предстоящему занятию.

Расписание учебных занятий во ФГБОУ ВО «ТГУ» создается отделом учебно-методического управления и диспетчерской службой, а затем утверждается заместителем ректора по развитию учебного процесса. Во время составления расписания занятий используется технология «Галактика Расписание учебных занятий», которая является дополнением ERP-системы «Галактика», позволяющая исключать ошибки, связанные с человеческим фактором, во время создания расписания. Эта часть процесса составления

расписания слишком сложна и велика, что в рамках данного проектирования рассматриваться не будет. Таким образом, данные, получаемые вследствие работы системы «Галактика Расписание учебных занятий» выступают в роли входных данных. Для публикации расписания учебных занятий, в едином информационном пространстве ФГБОУ ВО «ТГУ», экспортируются в MS Excel сотрудником диспетчерской службы. Расписание составляется и публикуется на 2 недели вперёд на сайте университета.

Таким образом, исходя из всего вышесказанного, можно сделать вывод, что подобная система публикации расписания не является удобной и даёт возможность для появления альтернатив. В данной дипломной работе будет рассмотрена технология публикации расписания занятий, которая является частью организации учебного процесса ФГБОУ ВО «ТГУ», и на основании полученной технологии затем должна быть создана система публикации расписания занятий в ФГБОУ ВО «ТГУ».

## **1.2 Анализ систем публикации расписания учебных занятий**

Системы публикации расписания учебных занятий служат для координирования студентов, преподавателей и сотрудников высших учебных заведений во времени. Они позволяют точно планировать свои действия. Рассмотрим публикацию расписания занятий в Московском государственном университете [3]. Зайдя на сайт <http://cacs.law.msu.ru/timeTable/student> студент может:

1. Выбрать факультет.
2. Указать вид подготовки.
3. Выбрать курс.
4. Выбрать группу.
5. Выбрать фамилию и инициалы студента.
6. Выбрать поиск по фамилии.
7. Указать диапазон дат.
8. Распечатать расписания.

По итогу на выходе система выведет ему расписание занятий конкретно для него или же для его группы.

На рисунке 1 показана диаграмма деятельности получения расписания учебных занятий в МГУ.

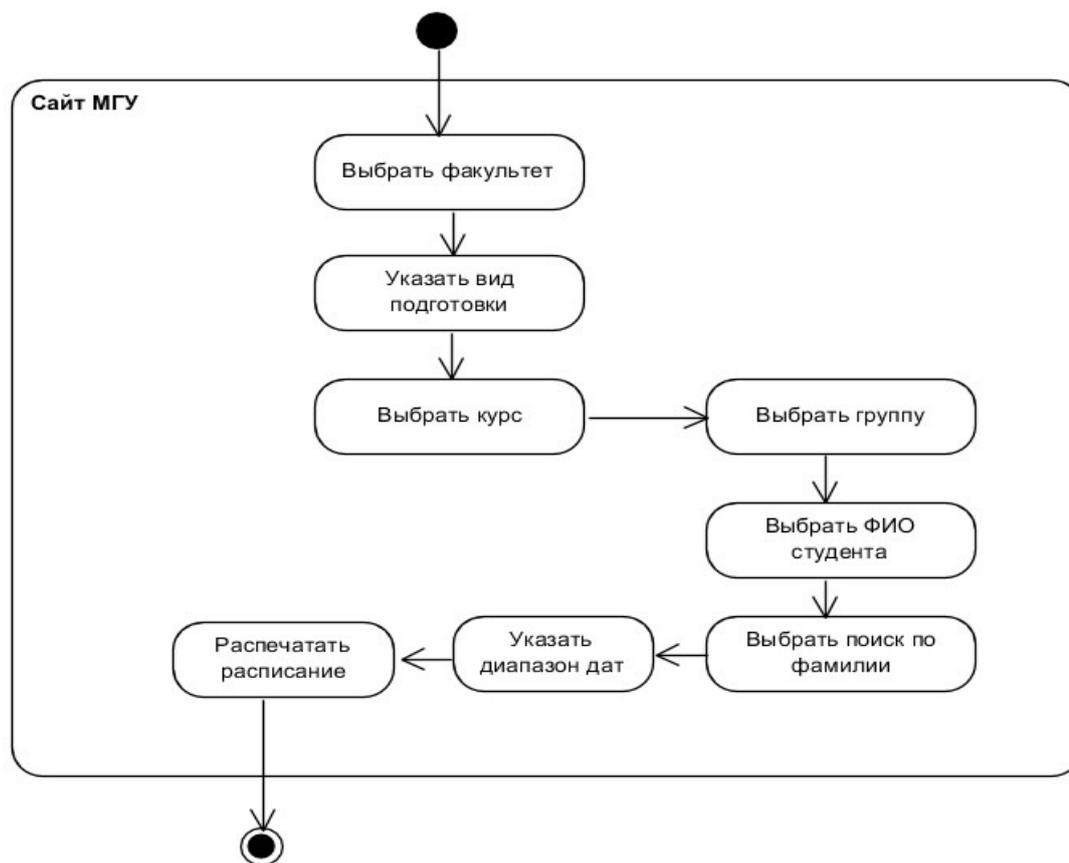


Рисунок 1 — Диаграмма деятельности получения расписания учебных занятий в МГУ

Во ФГБОУ ВО «ТГУ» реализована следующая модель публикации расписания: создан сайт университета, на котором публикуются табличные документы с данными о расписании занятий студентов. Вид документа представляет собой таблицу, в которой размещены данные о расписании занятий всего института по группам.

1. Открыть приложение браузер.
2. В адресной строке браузера набрать URL-адрес сайта ФГБОУ ВО «ТГУ» <https://www.tltsu.ru>.
3. На главной странице сайта перейти по ссылке «Студентам».

4. На странице «Студентам» перейти по ссылке «Расписание занятий».
5. На странице «Расписание занятий» выбрать необходимую неделю и свой институт.
6. Скачать файл, в название которого входит название института и номер недели, и открыть его с помощью предустановленного приложения для просмотра табличных файлов.
7. Найти необходимый курс обучения.
8. Найти необходимую учебную группу.
9. Просмотреть расписание.

Как видно из текущей модели получения расписания, в ней есть ряд недостатков:

- для просмотра расписания нужно получать файлы с сайта;
- чтобы найти требуемое расписание нужно выполнить 9 действий;
- не на каждом мобильном устройстве или компьютере, есть программы для открытия документа с расписанием;
- с точки зрения преподавателей и кафедр, нет возможности быстро узнать свободные аудитории.

Проведя 20 замеров времени с помощью 3 разных людей и на 3 устройствах, для того чтобы просмотреть расписание занятий своей группы в среднем у человека тратится 40 секунд. А время на загрузку самого файла занимает менее 1 секунды.

Таким образом, резюмируя всё вышесказанное можно сделать вывод, что такая рядовая задача, как просмотр расписания учебных занятий во ФГБОУ ВО «ТГУ», отнимает много времени и является процессом неудобным для выполнения, уступая перед другими ВУЗами.

### **1.3 Анализ сервисов планирования событий**

Системы планирования времени служат для планирования своей занятости. Они позволяют создавать события на определенную дату, с возможностью предварительного уведомления о их наступлении. Для того,

чтобы преподавателям и студентам было проще планировать своё время, требуется произвести интеграцию такой системы в систему публикации расписания занятий. Самыми популярными, сейчас являются: Google календарь [4], Яндекс календарь [5] и календарь Windows Live [6]. Рассмотрим каждый из них.

Google календарь – имеет в открытом доступе прикладной программный интерфейс(API), обладает собственными библиотеками для импорта данных.

Яндекс календарь – обладает закрытым прикладным программным интерфейсом, существует только возможность импорта данных в формате ICS.

Windows Live – так же присутствует API, но обладает достаточно низкой функциональностью и не имеет собственных библиотек для импорта данных.

В таблице 1 приведен анализ систем планирования времени, оцениваться они будут по двухбалльной шкале.

Таблица 1 – Анализ систем планирования времени

Система планирования времени	Google календарь	Яндекс календарь	Windows Live
Наличие API	1	0	1
Библиотеки для импорта данных	1	0	0
Поддержка всех версий браузеров	1	1	0
Итого	3	1	1

Таким образом, основываясь на критерии интеграции данных, для реализации импорта данных с сервиса Google Calendar принято решение использовать библиотеки, предоставленные компанией Google.

## 1.4 Выработка требований к будущей системе публикации расписания занятий в ФГБОУ ВО «ТГУ» и модулю оповещения

Для разработки требований возьмем систему классификации требований FURPS+ [7].

Выявленные требования для новой технологии представлены в таблице 1.

Таблица 2 — Требования к системе публикации расписания и модуля оповещения

№	Требование	Статус	Полезность	Обоснование выбора
Functionality – Функциональные требования				
1	Отображение списка преподавателей, групп, аудиторий, кафедр, институтов и сведения по каждому из них.	Одобрённые	Критичное	В системе должно быть реализовано хранение информации о преподавателях, группах, аудиториях, кафедрах и институтах.
2	Отображение расписания и заметок со сторонних сервисов	Одобрённые	Важное	В системе должна быть реализована возможность интегрировать данные из популярных приложений планирования времени (календарей) для комплексного планирования времени
3	Аутентификация пользователей	Одобрённые	Важное	В системе должна быть реализована функция аутентификации пользователей

Продолжение таблицы 2

4	Возможность оставлять заметки в календаре событий	Одобрённые	Важное	В системе должна быть реализована функция создания заметок в календаре событий
Usability – Требования к удобству использования				
6	Дизайн системы	Одобрённые	Важное	Дизайн системы должен быть

		нные		неброским, выполненным в удобной для восприятия палитрой цветов
Reliability – Требования к надежности				
6	Реализованная программа должна корректно реагировать на неверные данные и ошибки	Предложенные	Важное	Система должна анализировать данные на наличие ошибок и уметь обрабатывать их.
Performance – Требования к производительности				
7	Обработка запроса одного пользователя не должна превышать 1 секунду	Одобрённые	Критичное	Система должна иметь минимальную по времени реакцию для ответа на действия пользователя.
Supportability - Требование к поддержке				
8	Оформлять документацию по одной из популярных систем автоматической генерации проектной документации	Одобрённые	Важное	Система должна иметь документацию для быстрого ввода в курс дела новых людей в проекте.
Implementation requirements – Требования к реализации				
9	Система должна быть легко переносимой	Предложенные	Важное	Система должна быть реализованная в таком виде, чтобы ее можно было легко переносить.
Interface requirements – Требование к интерфейсу				

Продолжение таблицы 2

10	Получение данных с сторонних сервисов планирования событий	Одобрённые	Критичное	Система должна получать данные с сторонних систем планирования событий
<b>Physical requirements – Физические требования</b>				
11	Наличие интернет соединения	Одобрённые	Критичное	У системы должно присутствовать наличие интернет соединения для корректной работы и выкачки данных

Из функциональных требований самыми важными являются показ расписания и интеграция с сервиса планирования событий. Из нефункциональных самыми приоритетными являются наличие интернет соединения, и требования к тому, что обработка одного запроса не должна превышать более 1 секунды.

### **1.5 Постановка задачи на разработку системы публикации расписания занятий с модулем оповещения**

Целью работы является разработка системы публикации расписания учебных занятий с модулем оповещения. Система должна отвечать всем важным требованиям, выдвинутым в разделе 1.4 по системе классификации требований FURPS+.

Задачами на разработку системы публикации расписания учебных занятий с модулем оповещения являются:

1. Разработка новой технологии системы публикации расписания занятий с модулем оповещения.
2. Разработка архитектуры системы публикации расписания с модулем оповещения.
3. Разработка диаграммы последовательности по прецеденту "Получение расписания группы".
4. Разработка диаграммы классов системы публикации расписания занятий с модулем оповещения.

5. Разработка модуля оповещения.
6. Разработка математической модели системы публикации расписания и модуля оповещения.
7. Выбор средств реализации системы публикации расписания занятий с модулем оповещения.
8. Выбор интегрированной среды разработки, web-сервера приложений.
9. Описание основных принципов работы реализуемой системы публикации расписания занятий с модулем оповещения.
10. Разработка алгоритма классификации событий.

По итогу в этой главе был произведен анализ публикации расписания ТГУ, сравнение с другим институтом, анализ систем планирования времени, а так же произведена выработка требований к будущей системе и осуществлена постановка задачи на разработку системы публикации расписания занятий с модулем оповещения.

## 2 ПРОЕКТИРОВАНИЕ СИСТЕМЫ ПУБЛИКАЦИИ РАСПИСАНИЯ И МОДУЛЯ ОПОВЕЩЕНИЯ

### 2.1 Разработка новой технологии системы публикации расписания занятий с модулем оповещения

Диаграмма вариантов использования используется в данной выпускной квалификационной работе для того, чтобы смоделировать функциональные требования к проектируемой системе.

Актеры:

1. Пользователь — люди, использующий систему публикации расписания такие, как студенты, преподаватели, методисты.

В таблице 3 приведено описание прецедентов проектируемой системы публикации расписания занятий и модуля оповещения.

Таблица 3— Описание прецедентов проектируемой системы публикации расписания занятий и модуля оповещения

ID	Название прецедента	Описание
1	Получение и просмотр списка институтов преподавателей, аудиторий, групп, списка групп в институте, кафедр аудиторий, преподавателей, аудиторий на кафедре, преподавателей на кафедре, расписания аудиторий, преподавателей, групп, предмета, кафедры, поиск групп, преподавателей, аудиторий	функция, которая будет выводить список институтов преподавателей, аудиторий, групп, списка групп в институте, кафедр аудиторий, преподавателей, аудиторий на кафедре, преподавателей на кафедре, расписания аудиторий, преподавателей и групп, а так же высылать запрос на список найденных преподавателей, аудиторий и групп

Продолжение таблицы 3

2	Просмотр заметок со сторонних систем планирования времени	функция, которая будет импортировать и выводить данные со сторонних систем планирования времени
3	Импорт данных со сторонних систем планирования времени	функция, которая будет импортировать данные со сторонних систем планирования времени
4	Регистрация пользователей	функция, которая будет осуществлять регистрацию пользователей
5	Создание оповещения пользователем в системе публикации расписания занятий	функция, которая будет создавать оповещения в системе публикации расписания занятий

В таблице 4 представлено описание прецедента “Получение расписания групп”.

Таблица 4 – Описание прецедента «Импорт данных со сторонних систем планирования событий»

Прецедент: Получение расписания групп
ID: 3
Краткое описание: Реализация функции импорта данных со сторонних сервисов планирования событий
Главные актеры: 1. Пользователь
Второстепенные актеры:
Предусловие: Прецедент начинается по инициативе пользователя
Основной поток: 1. Пользователь запрашивает данные с сервиса планирования событий 2. Веб-клиент загружает данные в память 3. Веб-клиент преобразовывает данные 4. Веб-клиент выводит его события пользователю
Постусловие: Нет
Альтернативные потоки: Нет

В таблице 5 представлено описание прецедента “Создание оповещения пользователем в системе публикации расписания занятий”.

Таблица 5 — Описание прецедента «Создание оповещения пользователем в системе публикации расписания занятий»

Прецедент: Получение
ID: 5
Краткое описание: Функция, которая будет создавать оповещения
Главные актеры: 1. Пользователь
Второстепенные актеры:
Предусловие: Прецедент начинается по инициативе пользователя
Основной поток: 1. Система считывает данные, введенные с панели пользователя 2. Система отправляет полученные данные на сервер 3. Сервер записывает данные в базу данных 4. Система создает оповещение для пользователя
Постусловие: Нет
Альтернативные потоки: Нет

Диаграмма прецедентов отображает функциональные возможности проектируемого модуля для публикации расписания. Например, актер «пользователь» инициирует вариант использования «Получение списка институтов преподавателей», целью которого является получение данных о том, какие преподаватели есть в конкретном институте.

На рисунке 2 представлена диаграмма вариантов использования проектируемой системы.

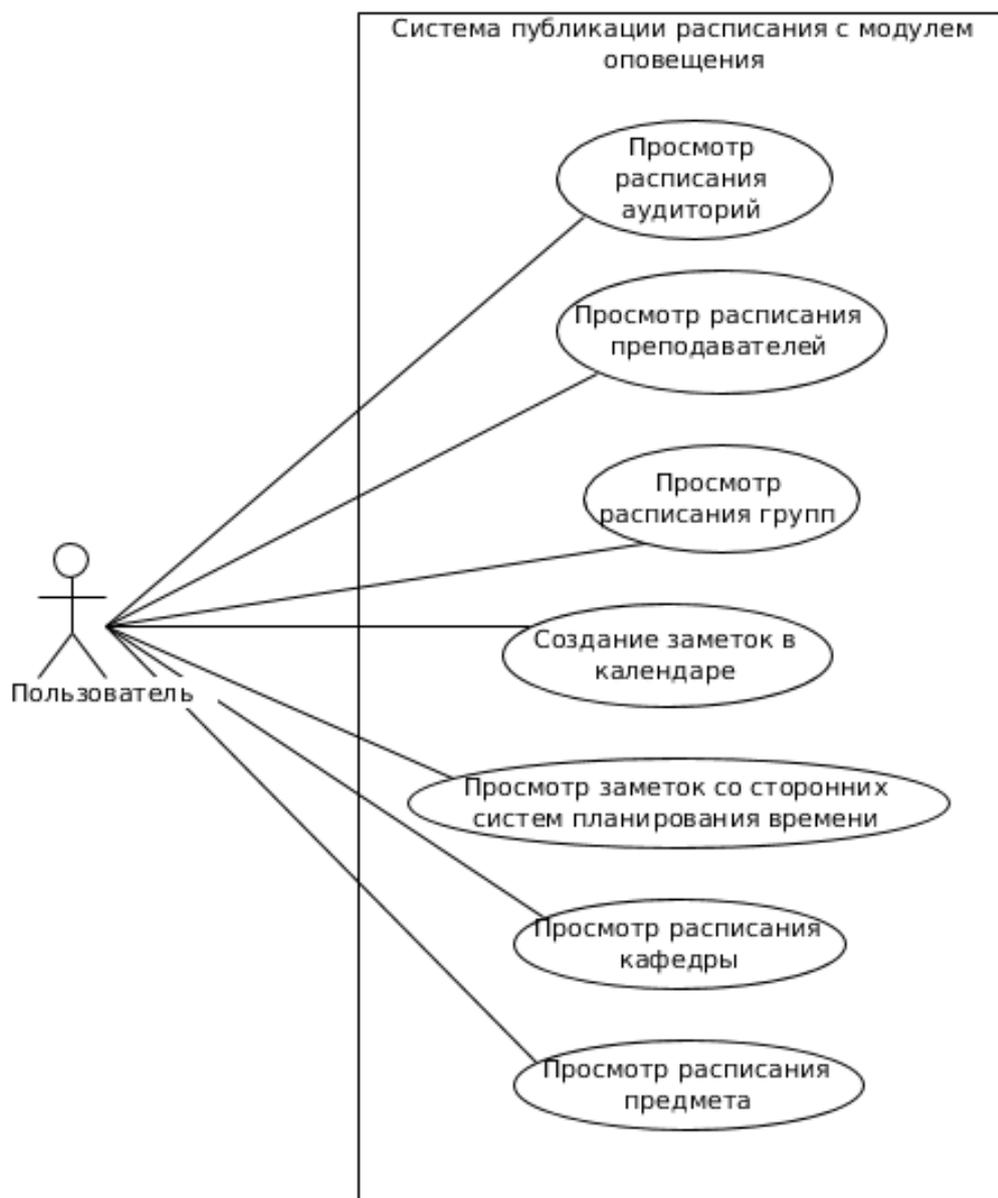


Рисунок 2 – Диаграмма прецедентов системы публикации расписания с модулем оповещения

Таким образом, в данном пункте были составлены основные прецеденты, описаны самые важные из них, выделен 1 актер, и составлена диаграмма прецедентов, в которой было показано, как прецедент взаимодействует с актером.

## 2.2 Разработка архитектуры системы публикации расписания с модулем оповещения

Система публикации расписания с модулем оповещения будет реализовано посредством того, что пользователь будет получать данные из

нескольких источников: web-сервер приложений системы публикации расписания ФГБОУ ВО «ТГУ» и сервисы планирования событий. Это , в свою очередь даёт возможность комбинирования событий из разных источников в единую модель публикации событий с возможностью их сортировки.

Система будет состоять из 4 компонентов:

1. Браузер пользователя.
2. Классы прикладного программного интерфейса системы публикации расписания учебных занятий ФГБОУ ВО «ТГУ».
3. Класс прикладного программного интерфейса Google календарь.
4. Компоненты СУБД.

В представленной на рисунке 3 диаграмме компонентов видно, что система имеет 2 внешних источника данных, это — API Google календарь и API расписания ФГБОУ ВО «ТГУ» которые работают через HTTP-протокол, а все функции по управлению данными реализуются в компоненте «Браузер». А через JDBC Driver в БД вносятся данные оповещений пользователя.

Диаграмма компонентов системы публикации расписания с модулем оповещения представлена на рисунке 3.

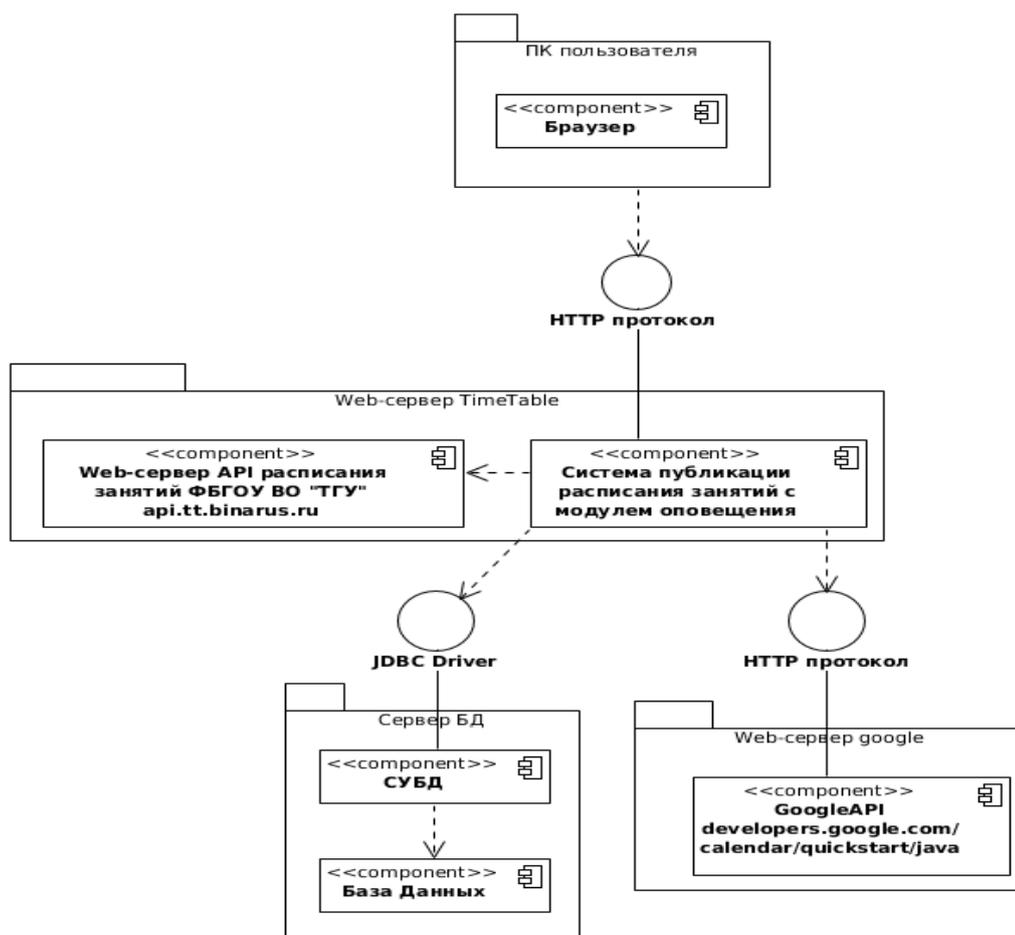


Рисунок 3 – Диаграмма компонентов проектируемой системы

Таким образом, в данном подразделе была разработана архитектура системы публикации расписания учебных занятий с модулем оповещения с помощью диаграммы компонентов.

### 2.3 Разработка диаграммы последовательности по прецеденту “просмотр расписания группы”

В соответствии с функциями системы необходимо реализовать диаграмму последовательности, функции построения таблицы расписания. Для того, чтобы система имела актуальные данные, ей необходимо их получать посредством связи с сервером приложений, на котором реализован API системы публикации расписания учебных занятий. Для того, чтобы просмотреть расписание группы, системе нужно выполнить следующий алгоритм действий:

1. Вызывается функция `getGroups()`.

2. Запрашивает данные json-файла у сервера приложений.
3. Преобразовывает и выводит данные в пригодный для публикации вид.
4. После выбора требуемой для пользователя группы вызывается функция getRaspTable().
5. Она запрашивает json-файла данные расписания у сервера приложений.
6. Преобразовывает и выводит данные в виде html таблицы.

Построенная диаграмма последовательности отображает динамический аспект объекта автоматизации и позволяет определить порядок реализации функций, что в свою очередь позволяет разработчику закодировать систему так, как она была задумана.

Просмотр расписания группы пользователем представлена на рисунке 4.

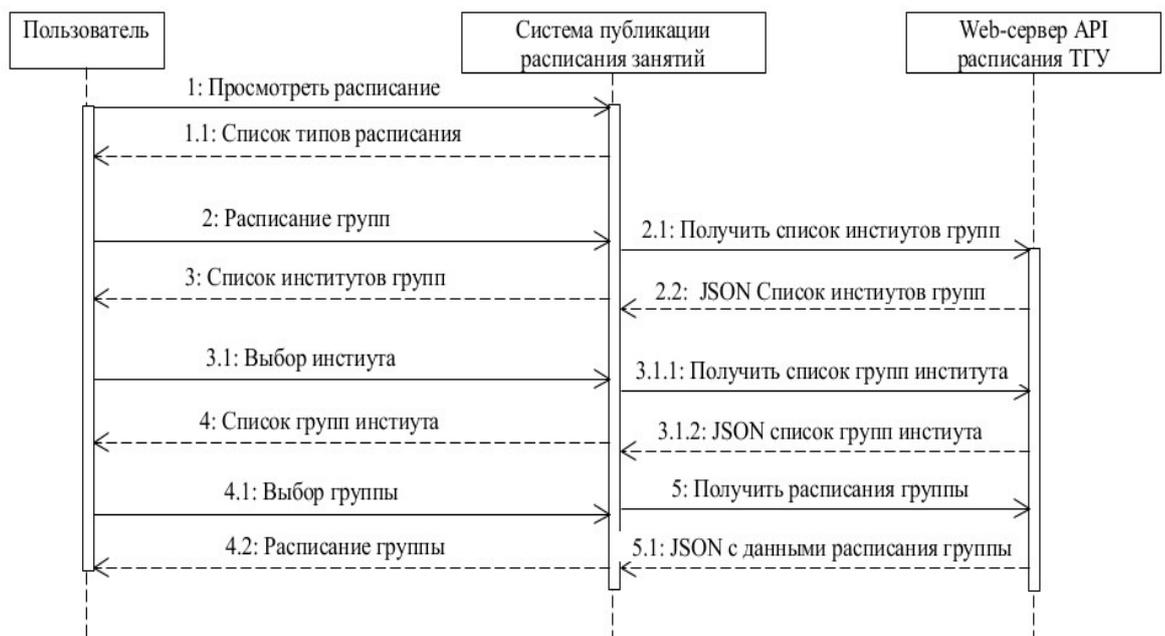


Рисунок 4 – Диаграмма последовательности по прецеденту “Просмотр расписания групп”

Таким образом, в данном подразделе были разработаны взаимодействия между элементами системы публикации расписания посредством диаграммы последовательности.

## 2.4 Разработка диаграммы классов системы публикации расписания занятий с модулем оповещения

Разработка вариантов использования позволила выделить классы и определить отношения между ними в соответствии с особенностями предметной области, в данном случае системы публикации расписания учебных занятий с модулем оповещения. Теперь требуется сформировать классы, методы и определить их роль в проектируемой системе публикации расписания с модулем оповещения.

В таблице 6 представлены классы и их роли в проектируемой системе.

Таблица 6 - Описание ролей классов в системе

Название класса	Роль класса в системе
Main	Скачивает json файлы, парсит, создает коллекцию и html таблицу
Rasp	Содержит описание сущности расписания

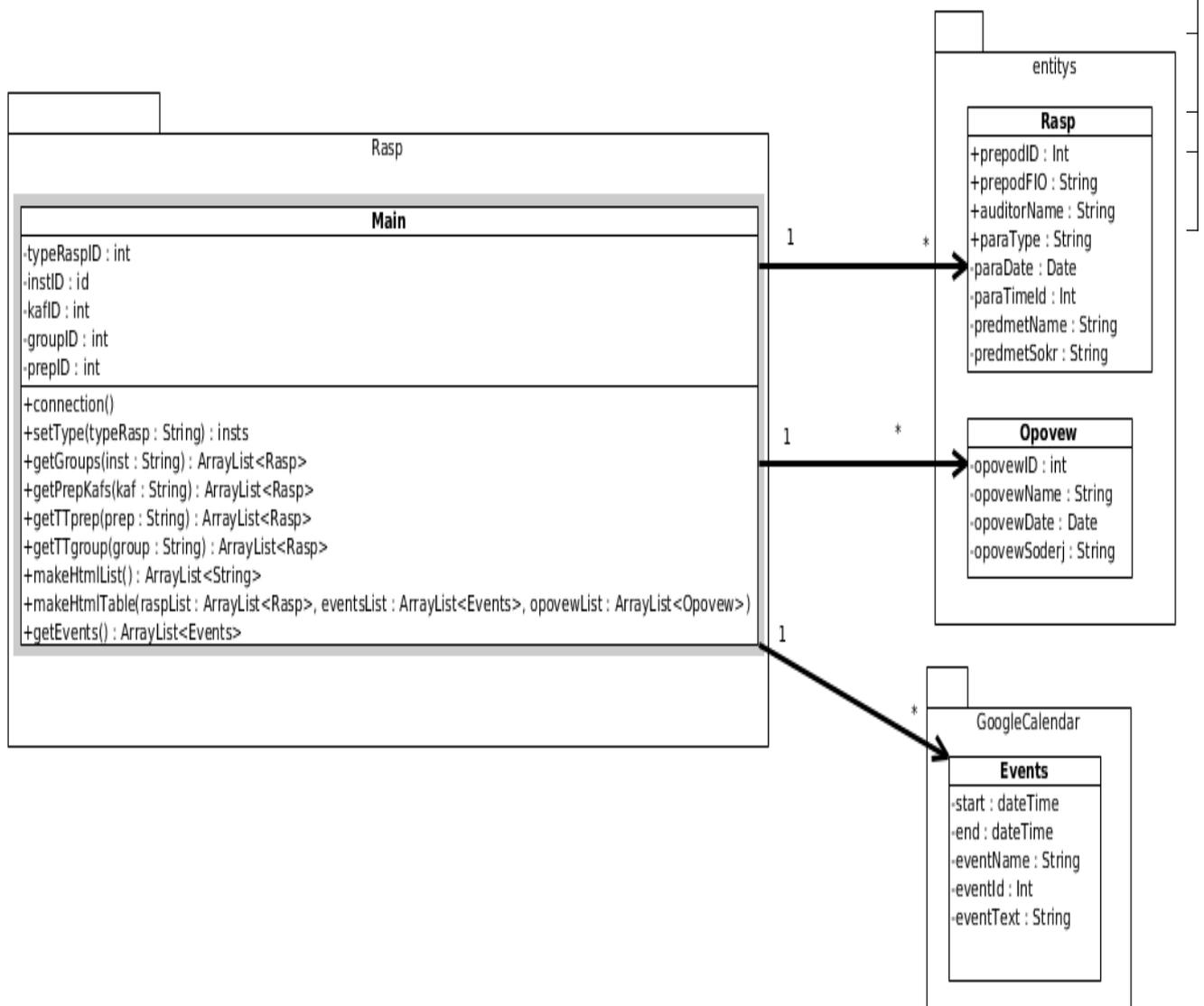


Рисунок 5 - Диаграмма классов проектируемой системы

В таблице 7 представлена спецификация класса Main.

Таблица 7 — Спецификация класса Main

Именованние	Назначение
connection	Создает соединение с web-сервером приложений
setType	Метод выбора получения расписания

#### Продолжение таблицы 7

getGroups	Получает список групп по кафедрам
getPrepkafs	Получает список преподавателей по кафедрам
getTTPrep	Получает расписание преподавателя
getTTgroup	Получает расписание группы
makeHtmlList	Создает и выводит списки
makeHtmlTable	Создает и выводит html таблицу с расписанием
getEvents	Получение событий с сервисов планирования событий

Класс Main содержит методы по получению и выводу данных расписания занятий и обеспечивает получение событий с сервисов планирования событий.

Таким образом, были разработаны основные классы и методы для реализации системы публикации расписания и модуля оповещения.

### 2.5 Разработка модуля оповещения

В систему публикации расписания занятий требуется реализовать модуль оповещения. Он позволит пользователям системы удобно планировать свое время, а так же, ввиду интеграции сервисов планирования событий, пользователь наглядно видит свои запланированные действия вне учёбы и расписание учебных занятий.

Для корректного отображения списка событий во времени данные требуется распределить во времени, чтобы избежать некорректного вывода данных. Полученные данные система будет преобразовывать в один единый вид представления данных.

Структура полей события будет следующая:

- название события;
- дата начала события;
- время начала события;
- дата окончания события;

-время окончания события;

-место события;

-описание события;

-приоритет.

Для более эффективного распределения времени, для оповещений будет такой атрибут как “приоритет”. Это позволит пользователям системы проще ориентироваться в планировании своего времени, и знать какие события нельзя пропускать. Но т.к. каждый раз указывать приоритет оповещения каждому событию отнимает достаточно много времени, этот процесс нужно автоматизировать. Система сама будет классифицировать события по степени важности, и выставлять им приоритет, который при необходимости можно изменить вручную.

Для того, чтобы создать оповещения в системе публикации расписания занятий, требуется выполнить следующий алгоритм действия:

1. Пройти аутентификацию в системе.
2. Выбрать институт.
3. Выбрать группу.
4. Выбрать день и номер занятия.
5. Создать оповещение.

На рисунке 6 показана диаграмма деятельности создания оповещения.



Рисунок 6 – Диаграмма деятельности создания оповещения

Таким образом, модуль оповещения позволит быстро информировать пользователей о предстоящих событиях, а приоритет этого события должна будет классифицировать система.

## **2.6 Обоснование и разработка математической модели классифицирования событий**

Для реализации автоматического классифицирования события по степени важности и присваиванию ему приоритета, требуется использовать один из алгоритмов классификации данных.

Выбор алгоритма классификации данных. Так как система будет автоматически расставлять приоритеты событий, лучше всего использовать алгоритм, который будет выполнять действия, исходя из выборки с уже заранее готовыми приоритетами. Для этого требуется использовать алгоритм, реализующий дерево принятия решений. Существуют несколько алгоритмов,

подходящих под эту задачу, самыми популярными являются ID3 [8] и CART [9]. Алгоритм CART имеет существенный минус - деревья решений, созданные алгоритмом, нестабильны. То есть результат, полученный на одной выборке, бывает не воспроизводим на другой. Ввиду этого был выбран алгоритм ID3.

Исходными данными задачи классификации является множество примеров событий  $T$ . Элементы этого множества событий описываются атрибутами полей, используемых для хранения данных в системе для планирования событий. Пусть метка класса принимает значения атрибута «приоритет»: важное, обычное, желательное.

Задача алгоритма заключается в построении дерева из множества примеров событий  $T$  из заранее подготовленной выборки событий. Процесс построения дерева будет происходить сверху вниз. Сначала создается корень дерева, а затем потомки корня и листья.

На первом шаге дерево не имеет узлов (только корень) и множество событий  $T$  (ассоциированное с корнем). Необходимо разбить исходное множество на подмножества. Для этого нужно выбрать один из атрибутов в качестве проверки. Тогда в результате разбиения получаются  $n$  (по числу значений атрибута, в данном случае  $n=3$ ) подмножеств  $i$ , соответственно, создаются три потомка корня, которым ставится в соответствие свое подмножество, получаемое при разбиении множества событий  $T$ . Затем эта процедура рекурсивно применяется ко всем потомкам корня и т.д.

Очевидно, что из  $m$  (по числу атрибутов) возможных вариантов, требуется выбрать наиболее подходящий.

Допустим, мы имеем проверку  $X$  (в качестве проверки может быть выбран любой атрибут, например, наименование события – поле name), принимающая три значения:  $A_1$ -важное,  $A_2$ -обычное,  $A_3$ -желательное. Тогда разбиение  $T$  по проверке  $X$  даст нам подмножества  $T_1, T_2 \dots T_3$ , при  $X$  равном соответственно  $A_1, A_2 \dots A_n$ . Единственная доступная информация –

распределение классов во множестве  $T$  и его подмножествах, которые получаются при разбиении по  $X$ . Это и играет роль при определении критерия.

Пусть  $freq(C_r, I)$  – число объектов из обучающей выборки относящихся к классу  $C_r$ . В этом случае, вероятность того, что случайно выбранный объект  $I$  будет принадлежать классу  $C_r$ , равняется:

$$P = \frac{freq(C_r, I)}{|I|} \quad (1)$$

Где  $P$  – вероятность того, что выбранный объект из множества событий  $I$ , будет принадлежать классу  $C_r$ ,

$freq(C_r, I)$  – количество событий, относящихся к конкретному классу,

$C_r$  – выбранный атрибут приоритета,

$I$  – множество событий относящихся к  $C_r$ .

Далее требуется подсчитать количество информации, основываясь на количестве объектов того или иного класса, получившихся в узле дерева после разбиения исходного множества.

Для получения оценки среднего количества информации, необходимого для определения класса объекта из множества  $T$ , даёт выражение:

$$H(x) = - \sum_{i=1}^n p(i) \log_2 p(i) \quad (2)$$

Где  $H(x)$  – количество информации после разбиения,

$x$  – текущий атрибут классификации,

$n$  - общее количество элементов,

$i$  – номер элемента,

$p$  – вероятность того, что выбранный объект из множества событий  $n$ , будет принадлежать классу  $C_r$ .

Подставляем в эту выборку полученное значение для  $P$ :

$$Info(I) = - \sum_{r=1}^k \frac{freq(C_r, I)}{|I|} * \log_2 \left( \frac{freq(C_r, I)}{|I|} \right) \quad (3)$$

Где  $Info(I)$  – среднее количество информации, необходимое для классификации события,

$r$  – порядковый номер события во множестве  $T$ ,

$k$  – общее количество событий в выборке,

$freq(C_r, I)$  – количество событий, относящихся к конкретному классу,

$C_r$  – один из атрибутов приоритета,

$I$  – множество событий выборки.

Ту же оценку, но уже после разбиения множества  $T$  по  $X_h$ , дает следующее выражение:

$$Info_{X_h}(T) = \sum_{i=1}^m \frac{T_i}{|T|} \cdot Info(T_i) \quad (4)$$

Где  $Info_{X_h}$  – оценка количества информации после разбиения,

$i$  – идентификатор события,

$m$  – общее количество событий в узле,

$T_i$  – событие выборки с номером  $i$ ,

$T$  – множество событий выборки,

$Info(T_i)$  – среднее количество информации, необходимое для классификации события  $i$ .

Тогда критерием для выбора атрибута будет являться формула:

$$Gain(X_h) = Info(I) - Info_{X_h}(I) \quad (5)$$

Где  $Gain(X_h)$  – критерий для выбора атрибута, выраженный в битах,

$X$  – текущий атрибут классификации,

$Info(I)$  – среднее количество информации, необходимое для классификации события,

$Info_{X_h}$  – количество информации, получаемое при выборе текущего атрибута.

Критерий (5) считается для всех атрибутов. Выбирается переменная с максимальным значением Gain. Этот атрибут станет проверкой в текущем узле, а потом по этому атрибуту будет производиться дальнейшее построение дерева.

Если в процессе работы алгоритма получен узел, ассоциированный с пустым множеством (ни один объект не попал в данный узел), то он помечается как лист, и в качестве решения листа выбирается наиболее часто встречающийся класс у непосредственного предка данного листа.

Таким образом, была разработана математическая модель алгоритма классификации событий из выборки для системы публикации расписания занятий с модулем оповещения.

В этой главе были построены диаграммы прецедентов, компонентов, последовательности и классов. Описаны основные функции проектируемой системы, описано преобразование данных в единый вид, модуль оповещений, а так же показана математическая модель алгоритма классификации событий по готовой выборке.

## **3 РЕАЛИЗАЦИЯ СИСТЕМЫ ПУБЛИКАЦИИ РАСПИСАНИЯ И МОДУЛЯ ОПОВЕЩЕНИЯ**

### **3.1 Выбор средств реализации системы публикации расписания занятий с модулем оповещения**

Перед тем как приступить к реализации, требуется выбрать язык программирования. Для этого подходят два популярных языка программирования это PHP [9] и Java [10]. PHP - скриптовый язык, интенсивно применяющийся для разработки динамических web-сайтов и web-приложений. Java – объектно-ориентированный язык программирования, где выполнение кода осуществляется с помощью виртуальной Java-машины. Для того, чтобы выбрать какой язык использовать, требуется провести сравнительный анализ этих двух языков. Оцениваться функции и возможности будут по десятибалльной шкале. Данные для анализа получены из источников [15][16][18]. Сам сравнительный анализ предоставлен в таблице 7.

Таблица 7 - Сравнительный анализ языков программирования

Функции и возможности	Языки программирования	
	Java	PHP
Практичность	8	5
Производительность	7	5
Библиотеки	6	6
Отладка	9	0
Простота изучения	8	8
Ресурсоемкость	5	8
Опыт работы	8	5
Итого:	51	37

Таким образом, при проведении анализа был выбран язык программирования – Java. Так же язык Java обладает большим количеством библиотек и достаточно прост в использовании. У него присутствует отладчик,

и он обладает большей производительностью. А так же, ввиду того, что опыта работы с данным языком больше, рациональней будет использовать именно его.

### **3.2 Выбор интегрированной среды разработки и web-сервера приложений**

Т.к. языком программирования было принято решение использовать Java, то теперь необходимо выбрать среду разработки. Самыми популярными являются Eclipse [12] и Netbeans [13]. Eclipse – кроссплатформенная интегрированная среда разработки модульных приложений. Netbeans – свободная кроссплатформенная интегрированная среда разработки. Для того, чтобы выбрать более подходящую среду разработки, проведем выбор по следующим критериям:

- производительность — к производительности можно отнести, насколько быстро у IDE скорость сборки проектов, их обработка и компиляция;
- совместимость плагинов — к совместимости относится то, насколько корректно работают плагины в данных IDE;
- обновление документации - то, насколько своевременно обновляется документация к данным средам разработки;
- локализация — к локализации относится то, обладает ли IDE встроенным русским языком или сколько времени потребуется на установку.

В таблице 8 приведен анализ и выбор интегрированной среды разработки.  
Таблица 8 — Выбор интегрированной среды разработки

Функции и возможности	Интегрированная среда разработки	
	Eclipse IDE	Netbeans IDE
Производительность	9	10
Совместимость плагинов	5	8

Продолжение таблицы 8

Обновление документации	5	8
Локализация	3	9

Итого:	22	35
--------	----	----

Обе среды разработки обладают своими плюсами и минусами. Но ввиду того, что Eclipse уступает перед Netbeans производительностью, проблемой совместимости плагинов и отсутствием русификации, было принято решение выбрать Netbeans. Так же в Eclipse нет своевременного обновления документации, в отличии от Netbeans, у которого есть поддержка в виде независимого сообщества разработчиков-энтузиастов.

После того как выбран язык и интегрированная среда разработки требуется выбрать web-сервер приложений, где будет работать программа. Web-сервер приложений - это программная платформа, предназначенная для эффективного исполнения программ и скриптов, на которых построены приложения. Основная задача компонентов сервера для веб приложений — обеспечивать создание динамических страниц. Но современные серверы приложений включают в себя и балансировку нагрузки и поддержку кластеризации, а так же повышенную отказоустойчивость, позволяя таким образом разработчикам сфокусироваться только на реализации бизнес-логики.

Для реализации был выбран web-сервер приложений WildFly [14] (Jboss As) - Java EE сервер с открытым исходным кодом. Данный web-сервер приложений имеет полномасштабный сервер приложений Java EE, включая контейнер EJB, имеет более широкое и более глубокое сообщество пользователей и более зрелую кодовую базу, нежели другие Java EE web-серверы приложений. Данный web-сервер бесплатный — его можно свободно загрузить и использовать. Так же он обладает простой установкой и настройкой.

Таким образом был сделан выбор интегрированной среды разработки Netbeans и web-сервера приложений WildFly.

### **3.3 Описание основных принципов работы реализуемой системы публикации расписания занятий с модулем оповещения**

Основная задача реализуемого программного обеспечения - это создание системы публикации данных расписания, органайзера и создание оповещений.

Выделим основные функции системы:

- загрузка данных в формате json;
- преобразование данных из json в пригодную для вывода форму;
- вывод данных в виде html таблицы;
- интеграция сервисов планирования событий;
- возможность создания оповещений;
- классификация приоритетов событий.

Реализация функции загрузки данных в формате json с данными расписания учебных занятий. Реализация классов находится в пакете `gasp`. Это основной пакет в системе, он контролирует работу системы, в нём содержатся основные классы. Эти классы отвечают за загрузку и обработку данных формата json, запись данных в объекты класса, и в дальнейшем преобразование их в форму, необходимую для вывода данных, и непосредственно сам вывод данных.

Все эти классы работают по похожему алгоритму. Единственная разница состоит в том, что они используют данные из разных файлов.

Алгоритм реализации классов пакета `gasp`:

- открытие сетевого соединения с сервером, откуда выкачиваются данные в формате json;
- обработка данных;
- приведение их в необходимую для отображению форму;
- запись данных в объекты класса;
- формирование данных в виде html-таблицы;
- вывод данных.

Далее происходит реализация обработки полученных данных стандартными средствами работы с данными в json формате. Все поля, относящиеся к данному виду, имеют постфиксы, указывающие на тип данных. Например все поля, оканчивающиеся на “Name” являются указателем на название содержимого, хранимого в поле, а поля с постфиксом “Id” хранят в себе идентификатор объекта в json формате..

Для того, чтобы преобразовать данные из формата json в поля, необходимо создать JSONObject, в который помещается содержимое полученного файла и получить доступ к его полям и их значениям через их имена.

Полученные данные помещаются в объекты коллекции ArrayList и могут быть выведены на экран или обработаны в дальнейшем. Так как ArrayList использует массив, то время доступа к элементу по индексу минимально. При удалении произвольного элемента из списка все элементы, находящиеся «правее», смещаются на одну ячейку влево, при этом реальный размер массива не изменяется. Если при добавлении элемента оказывается, что массив полностью заполнен, будет создан новый массив размером  $(n * 3) / 2 + 1$ , в него будут помещены все элементы из старого массива + новый добавляемый элемент.

Для того, чтобы выводить данные нужно создать html таблицу, в которой должна быть определена её ширина, заголовки, заполнение данных и непосредственная сборка.

Например, на рисунке 7 отображена диаграмма функции получения расписания группы.

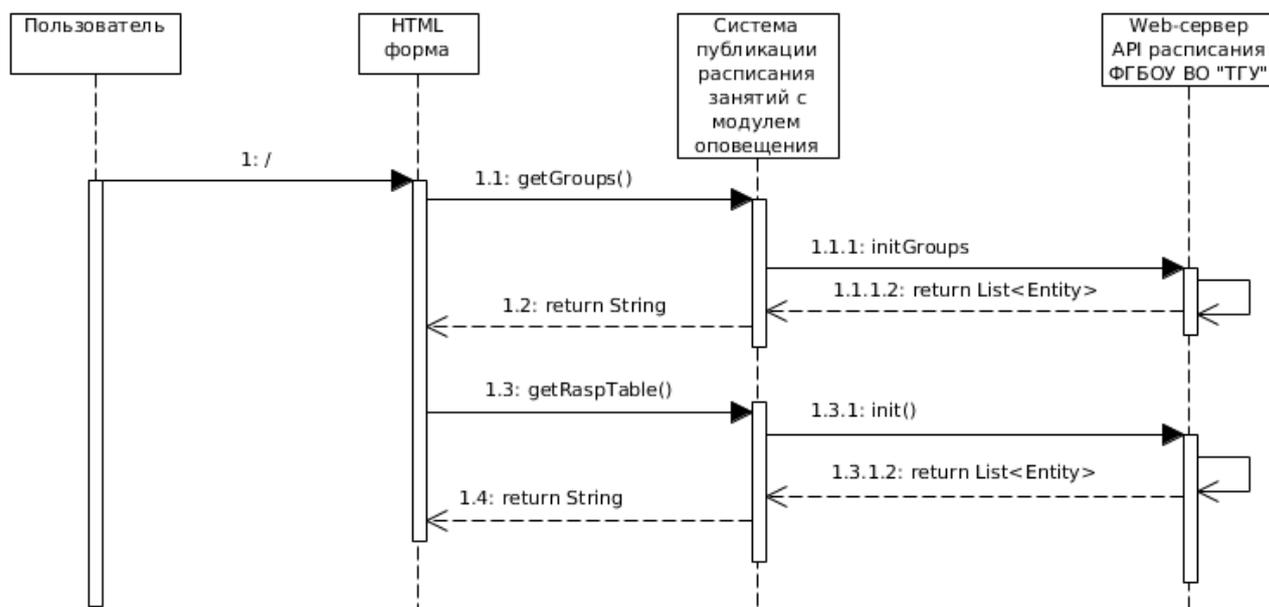


Рисунок 7 – диаграмма последовательности метода getGroups()

Таким образом, в данном пункте были показаны и описаны основные функции работы программы, по работе с данными в формате json, выводом данных, построением html-таблиц и созданием оповещений.

### 3.4 Разработка алгоритма классификации событий

Класс-классификатор работает по следующему алгоритму:

1. Создание корня дерева принятия решений.
2. Выбор атрибута классификации.
3. Разбиение элементов выборки по атрибуту «Приоритет».
4. Оценка качества разбиения (значение Gain).
5. Повторение процедуры до достижения максимального значения Gain.

На рисунке 8 показана схема работы алгоритма классификации по атрибуту “Приоритет”.



Рисунок 8 – Алгоритм метода класса ID3Classifier

В таблице 9 показан пример выборки до классификации.

Таблица 9 – выборка до классификации

№	Наименование	Дата	Время	Текст	Метка класса
1	Лекция	24.04.2018	13:15	Лекция по мат. анализу	
2	Тренировка	25.04.2018	19:00	Тренировка в спортзале	
3	Практические занятия	26.04.2018	8:30	Практические занятия в институте	

Продолжение таблицы 9

4	Зачет	27.04.2018	10:15	Зачет по физ. культуре	
5	Экзамен	28.04.2018	14:15	Экзамен по мат. анализу	
6	Магазин	29.04.2018	15:00	Поход в магазин	
7	Друзья	30.04.2018	18:00	Встреча с друзьями	

Далее классифицируется метка класса по следующей формуле

$$Gain(X_h) = Info(I) - Info_{X_h}(I) \quad (6)$$

Где  $Gain(X_h)$  – критерий для выбора атрибута, выраженный в битах,  
 $X$  – текущий атрибут классификации,

$Info(I)$  – среднее количество информации, необходимое для классификации события,

$Info_{X_h}$  – количество информации, получаемое при выборе текущего атрибута.

Далее в таблице 10 предоставлена выборка после классификации где в столбце “Метка класса” 0 – “обычное”, 1 – “желательное”, 2 – “важное”.

Таблица 10 – выборка после классификации

№	Наименование	Дата	Время	Текст	Метка класса
1	Лекция	24.04.2018	13:15	Лекция по мат. анализу	1
2	Тренировка	25.04.2018	19:00	Тренировка в спортзале	0
3	Практические занятия	26.04.2018	8:30	Практические занятия в институте	2
4	Зачет	27.04.2018	10:15	Зачет по физ. культуре	2

5	Экзамен	28.04.2018	14:15	Экзамен по мат. анализу	2
6	Магазин	29.04.2018	15:00	Поход в магазин	0
7	Друзья	30.04.2018	18:00	Встреча с друзьями	1

Таким образом, разработка алгоритма построения дерева принятия решений адаптирована под решение задачи классификации событий по атрибуту “Приоритет”, который работает до достижения максимального значения качества разбиения Gain.

### **3.5 Разработка диаграммы развертывания системы публикации расписания учебных занятий с модулем оповещения**

Разработанная система публикации расписания учебных занятий с модулем оповещения состоит из следующих элементов:

1. Браузер пользователя.
2. Система публикации расписания учебных занятий.
3. Web-сервер API расписания учебных занятий ФГБОУ ВО ТГУ.
4. СУБД.
5. База данных.
6. Web-сервер google.

Данные элементы взаимосвязаны между собой и в системе персональный компьютер пользователя взаимодействует с системой посредством HTTP протокола. Сама система взаимодействует с web-сервером google так же через HTTP протокол, а с сервером базы данных, она взаимодействует через JDBC драйвер.

Корректная работоспособность данной системы была выявлена на операционной системе Windows, версии 7.

На рисунке 9 показана диаграмма развертывания системы публикации расписания учебных занятия с модулем оповещения.

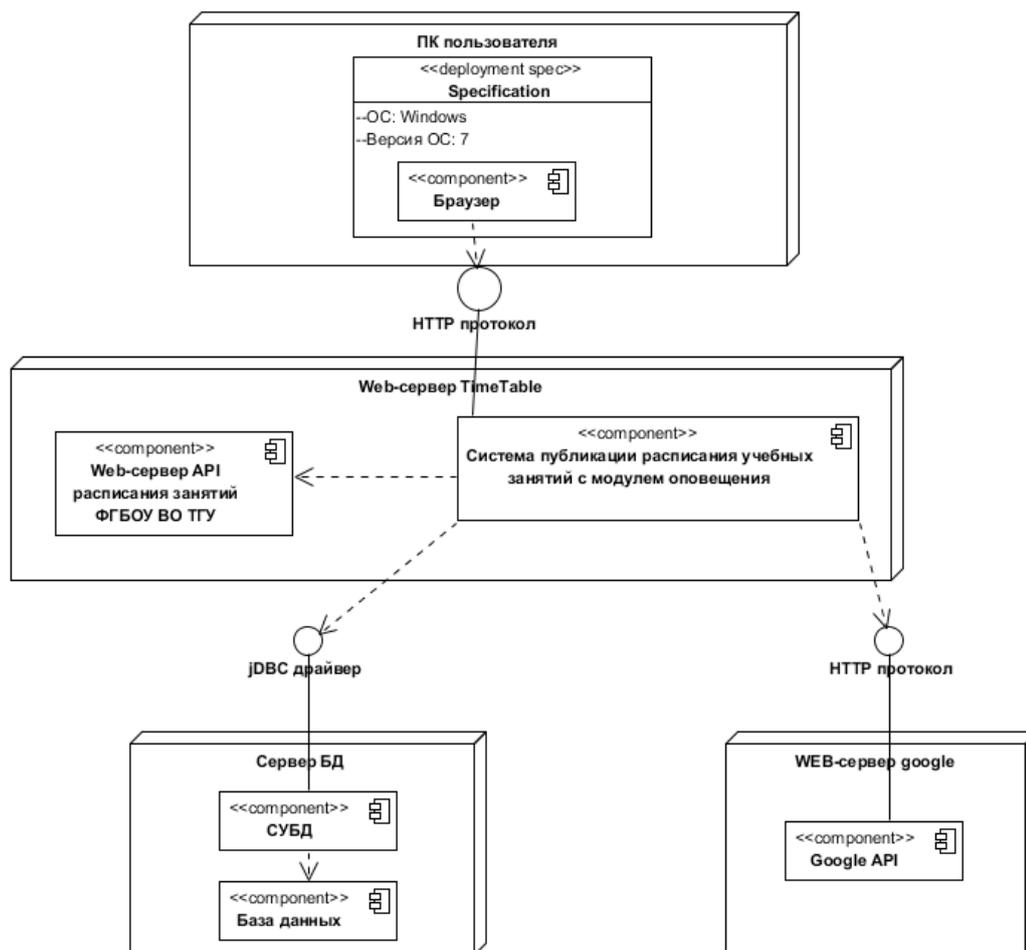


Рисунок 9 – диаграмма развертывания системы публикации расписания учебных занятия с модулем оповещения.

Таким образом, была разработана диаграмма развертывания системы публикации расписания учебных занятий с модулем оповещения.

В этой главе был произведен анализ и выбор языка программирования, интегрированной среды разработки, и web-сервера приложений. Так же были показаны и описаны основные функции разрабатываемой системы, разработан алгоритм классификации, приведен пример выборки, а так же была разработана диаграмма развертывания.

## **ЗАКЛЮЧЕНИЕ**

В ходе проведения работы в подразделе 1.1 была исследована публикация расписания учебных занятий в ФГБОУ ВО “ТГУ”.

В подразделе 1.2 был проведен анализ системы публикации расписания учебных занятий на примере Московского государственного университета.

В подразделе 1.3 был проведен анализ существующих систем планирования времени.

Анализ данных разделов выявил необходимость реализации собственной системы публикации расписания учебных занятий с модулем оповещения для более удобного планирования времени.

По системе классификации требований FURPS+, в разделе 1.4, были выработаны требования к разрабатываемой системе публикации расписания учебных занятий с модулем оповещения.

На основе требований к разрабатываемому алгоритму, в разделе 2.1, были выработаны требования к технологии системы публикации расписания занятий и модулю оповещения.

На основе выработанных требований к технологии системы публикации расписания занятий и модулю оповещения, в подразделе 2.2, была разработана архитектура системы публикации расписания учебных занятий с модулем оповещения.

На основе разработанной архитектуры системы публикации расписания с модулем оповещения, в подразделе 2.3, были разработаны взаимодействия между элементами системы посредством диаграммы последовательности.

На основании этого, в пункте 2.4 были разработаны основные классы и методы разрабатываемой системы публикации расписания занятий с модулем оповещения.

Таким образом, в подразделе 2.5, был разработан модуль создания списка событий в разрабатываемой системе.

На основании этого, в пункте 2.6, был разработан основной алгоритм модуля оповещения.

Исходя из этого, в подразделе 2.7, было описано обоснование математической модели алгоритма для классификации событий.

В ходе выбора средств реализации системы публикации расписания занятий с модулем оповещения, в подразделе 3.1, был определен язык программирования.

С учетом выбранного языка программирования, в подразделе 3.2, был сделан выбор интегрированной среды разработки и web-сервера приложений.

В подразделе 3.3 сделано описание основных принципов работы реализуемой системы публикации расписания занятий с модулем оповещения.

В подразделе 3.4 был разработан алгоритм классификации событий и приведен пример выборки.

В подразделе 3.5 была разработана диаграмма развертывания системы публикации расписания учебных занятий с модулем оповещения.

В ходе работы была разработана система публикации расписания учебных занятий с модулем оповещения.

## СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

### *Научная и методическая литература*

1. Фаулер М. Рефакторинг. Улучшение кода 2-е издание // М. Фаулер. – СПб : Символ Плюс, 2015. – 432с.

### *Электронные ресурсы*

2. Тольяттинский государственный университет [Электронный ресурс] // Тольяттинский государственный университет: История Тольяттинского государственного университета. – URL: <http://www.tltsu.ru> (дата обращения:20.12.2017)
3. АСУ Учебным процессом юридического факультета МГУ имени Ломоносова [Электронный ресурс] – URL: <http://cacs.law.msu.ru/timeTable/student> (дата обращения:20.12.2017)
4. Google календарь [Электронный ресурс] – URL: <https://www.google.com/calendar> (дата обращения:25.12.2017)
5. Yandex календарь [Электронный ресурс] – URL: <https://calendar.yandex.ru/> (дата обращения:25.12.2017)
6. Windows Live календарь [Электронный ресурс] – URL: <https://office.live.com/start/Calendar> (дата обращения:25.12.2017)
7. Требования к системе: классификация FURPS+ [ Электронный ресурс] – URL: <https://sysana.wordpress.com/2010/09/16/furps/> (дата обращения:10.01.2018)
8. Алгоритм ID3 [Электронный ресурс] – URL: [http://life-prog.ru/1\\_20468\\_algorithm-ID.html](http://life-prog.ru/1_20468_algorithm-ID.html) (дата обращения:07.03.2018)
9. Деревья решений — CART математический аппарат [Электронный ресурс] – URL: <https://basegroup.ru/community/articles/math-cart-part1> (дата обращения:07.03.2018)
- 10.Изучение языка программирования PHP [Электронный ресурс] – URL: <https://tproger.ru/tag/php/> (дата обращения:15.04.2018)

11. Всё для изучения JAVA [Электронный ресурс] – URL: <https://tproger.ru/tag/java/> (дата обращения:15.04.2018)
12. Работа в Eclipse IDE [Электронный ресурс] – URL: <http://study-java.ru/uroki-java/urok-3-znakomstvo-s-eclipse/> (дата обращения:16.04.2018)
13. Установка и настройка Netbeans IDE [Электронный ресурс] – URL: [https://geekbrains.ru/posts/netbeans\\_setup](https://geekbrains.ru/posts/netbeans_setup) (дата обращения:16.04.2018)
- 14.Установка сервера Wildfly [Электронный ресурс] – URL: <http://patchwork.deskbook.info/start-wildfly/9-wildfly-install.html> (дата обращения: 17.04.2018)
- 15.Википедия – свободная энциклопедия / [Электронный ресурс] – URL: <https://ru.wikipedia.org/wiki/>, Режим доступа: свободный. (дата обращения: 17.04.2018)
- 16.Выбор между PHP и JAVA / [Электронный ресурс] – URL: <https://habrahabr.ru/post/262461/>, Режим доступа: свободный (дата обращения: 17.04.2018)
- 17.IBM Knowledge Center / [Электронный ресурс] – URL: [https://www.ibm.com/support/knowledgecenter/ru/ssw\\_aix\\_61/com.ibm.aix.performance/advantages\\_java.htm](https://www.ibm.com/support/knowledgecenter/ru/ssw_aix_61/com.ibm.aix.performance/advantages_java.htm), Режим доступа: свободный. (дата обращения: 17.04.2018)
- 18.IDEF0 - методология функционального моделирования / Itstan [Электронный ресурс]. – URL: <http://www.itstan.ru/funk-strukt-analiz/idef0-metodologija-funkcionalnogo-modelirovanija.html/>, Режим доступа: свободный. (дата обращения: 17.04.2018)

*Литература на иностранном языке*

- 19.Baesens, B. Beginning Java Programming: The Object-Oriented Approach / B. Baesens, A. Backiel, S. Vanden Broucke. – 1st edition, Wrox, 2015.
- 20.Bill Burke, RESTful Java with JAX-RS 2.0, Second Edition / Bill Burke - O'Reilly Media, Inc. – 392 p. 1

21. David Skla. Learning PHP: A Gentle Introduction to the Web's Most Popular Language / David Skla. - O'Reilly, 2016. – 416c.
22. Deitel, H. Java How to Program / H. Deitel, P. Deitel. – 9th edition, Prentice Hall, 2015.
23. Hudson O. Getting started with IntelliJ IDEA // O. Hudson, Birmingham: Packt Publishing, 2013. – 114p.
24. Krochmalski J. IntelliJ IDEA Essentials // J. Krochmalski. – Birmingham: Packt Publishing, 2014.-263p.
25. Masoud Kalali, Developing RESTful Services with JAX-RS 2.0, WebSockets, and JSON / Masoud Kalali – United Kingdom, Birmingham,: Packt Publishing, LTD, 2013. – 107 p.

**Листинг кода реализации классов**

Прикреплен на диске