МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество

«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

им. ГУМАРБЕКА ДАУКЕЕВА»

Институт информационных технологии

Кафедра IT–инжиниринг

**ДОПУЩЕН К ЗАЩИТЕ**

Заведующий кафедрой

PhD, доцент

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.У. Утегенова

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

**ДИПЛОМНЫЙ ПРОЕКТ**

На тему: Разработка автоматизированной системы составления расписания для филиала г. Алматы НАО «Республиканская физико-математическая школа».

Специальность: 5В060200 – «Информатика»

Выполнил: Имашев Д.С. Группа: ИНФ-17-2

Научный руководитель: PhD, доцент Утегенова А.У.

Консультанты:

по экономической части: к.э.н., профессор \_\_\_\_\_\_\_\_\_\_\_\_\_\_ К.Р. Габелашвили

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

по безопасности

жизнедеятельности: д.т.н., ст. преп. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ш.Ш. Бекбасаров

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

по применению

вычислительной техники: ст. преп. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М.Н. Майкотов

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Нормоконтролер: ст. преп. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Б.Р. Абсатарова

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Рецензент:

к.т.н доцент, КНУ Савельева Е.А.

(учёная степень, звание, Ф.И.О.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2021 г.

Алматы 2021

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество

«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

им. ГУМАРБЕКА ДАУКЕЕВА»

Институт информационных технологии

Кафедра IT–инжиниринг

Специальность 5В060200 – «Информатика»

**ЗАДАНИЕ**

на выполнение дипломного проекта

Студенту Имашеву Даулету Сериковичу

Тема проекта: Разработка автоматизированной системы составления расписания для филиала г. Алматы НАО «Республиканская физико-математическая школа»

Утверждена приказом по университету № 25от «18»февраля2021 г.

Срок сдачи законченного проекта «\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2021 г.

Исходные данные к проекту (требуемые параметры результатов исследования (проектирования) и исходные данные объекта): Microsoft Visual Studio - среда разработки веб приложения, С# - язык программирования, MS SQL Server – база данных.

Перечень вопросов, подлежащих разработке в дипломном проекте, или краткое содержание дипломного проекта:

* аналитическая часть;
* проектная часть;
* экспериментальная часть;
* экономическая часть;
* безопасность жизнедеятельности;
* приложение А. Листинг программы.

Перечень графического материала (с точным указанием обязательных чертежей): представлены 15 таблиц, 30 иллюстраций, презентация (19 слайдов).

Основная рекомендуемая литература:

* 1. Биллиг, В.А. Основы программирования на С#: Учебное пособие / В.А. Биллиг. – М.: Бином, 2012. – 483 c.
  2. Скит. C# для профессионалов. Тонкости программирования / Джон Скит, 2019. – 608 с.
  3. Современные технологии программирования: разработка Windows-приложений на языке С# / С.В. Горелов, 2019 – 363 с.
  4. Г. Боканова Методические указания по выполнению экономической части дипломных работ Алматы, АУЭС, 2019 – 33с.

Консультация по проекту с указанием относящихся к ним разделов проекта

|  |  |  |  |
| --- | --- | --- | --- |
| Раздел | Консультант | Сроки | Подпись |
| Технико-экономическое обоснование проекта | PhD Габелашвили К.Р. |  |  |
| Безопасность жизнедеятельности | ст. преп. Бекбасаров Ш.Ш. |  |  |
| Программное обеспечение | ст.преп. Майкотов М.Н. |  |  |
| Нормоконтролер | ст.преп. Абсатарова Б.Р. |  |  |

ГРАФИК

подготовки дипломного проекта

|  |  |  |
| --- | --- | --- |
| Наименования разделов, перечень  разрабатываемых вопросов | Сроки представления  научному руководителю | Примечания |
| Анализ предметной области |  |  |
| Выбор программного обеспечения |  |  |
| Составление технического задания |  |  |
| Создание баз данных |  |  |
| Реализация программного продукта |  |  |

Дата выдачи задания «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.У. Утегенова

Научный руководитель проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.У. Утегенова

Задание принял к исполнению студент \_\_\_\_\_\_\_\_\_\_\_\_Д.С. Имашев

**Андатпа**

Дипломдық жоба «Республикалық физика-математика мектебі» КЕАҚ Алматы филиалы үшін жоспарлаудың автоматтандырылған жүйесін жасауға арналған. Жасалған бағдарлама оқу бөлімі басшысына келесі тоқсанға арналған кестені тез және оңай құруға көмектеседі. Бағдарлама C # бағдарламалау тілінде Microsoft Visual Studio интеграцияланған даму ортасында әртүрлі кітапханаларды қолдана отырып жүзеге асырылды. Бағдарлама интерфейсі Windows Forms технологиясының көмегімен терезедегі қосымша түрінде жасалған, онда санаттар бойынша әр түрлі қойындылар орналасқан.

Сонымен қатар, қорытынды жұмыста мұғалімдердің өтініштері талданады және барлығына ыңғайлы кесте құру мәселесі қарастырылады.

**Аннотация**

Дипломный проект посвящен разработке автоматизированной системы составления расписания для филиала г. Алматы НАО «Республиканская физико-математическая школа». Разработанная программа поможет заведующим учебной части быстро и без труда составлять расписание на ближайшую четверть. Программа была реализована на языке программирования C# в интегрированной среде разработки Microsoft Visual Studio с использованием различных библиотек. Интерфейс программы был создан с помощью технологии Windows Forms в виде оконного приложения, в котором располагаются различные вкладки по категориям.

Кроме того, в выпускной работе проводится анализ запросов преподавателей и рассматривается вопрос удобного составления расписания для каждого.

**Annotation**

The graduation paper is devoted to Automatic scheduling development for branch of NJSC «National school of physics and maths» in Almaty. The developed program will help the head of the educational unit quickly and easily draw up a schedule for the next quarter. The program was implemented in the C # programming language in the Microsoft Visual Studio integrated development environment using various libraries. The program interface was created using Windows Forms technology in the form of a windowed application in which various tabs are located by category.

In addition, the final work analyzes the requests of teachers and considers the issue of convenient scheduling for everyone.

**Содержание**

[Введение 8](#_Toc72366532)

[1 Аналитическая часть 9](#_Toc72366533)

[1.1 Анализ деятельности организации 9](#_Toc72366534)

[1.2 Организационная структура и система управления……………………….](#_Toc72366534)10

[1.3 Цели и задачи разработки 1](#_Toc72366535)1

[1.4 Обоснование выбора ПО 1](#_Toc72366536)1

[1.5 Спецификация и обоснование нефункциональных требований 12](#_Toc72366537)

[2 Проектная часть 1](#_Toc72366538)6

[2.1 Функциональная структура 1](#_Toc72366539)6

[2.2 Информационное обеспечение 1](#_Toc72366540)7

[2.3 Математическое обеспечение……………………………………………….21](#_Toc72366540)

[2.4 Программная часть](#_Toc72366541) 22

[2.5 Обеспечение информационной безопасности……………………………..](#_Toc72366540)33

[3 Экспериментальная часть 30](#_Toc72366542)

[3.1 Контольный пример 37](#_Toc72366543)

[4 Технико-экономическое обоснование проекта](#_Toc72366544) 44

[4.1 Определение объёма и трудоёмкости разработки веб-приложения](#_Toc72366545) 44

[4.2 Расчёт затрат на разработку веб-приложения](#_Toc72366546) 45

[4.3 Амортизация основных фондов (ОФ) 4](#_Toc72366547)8

[4.4 Смета затрат на разработку ПО 4](#_Toc72366548)9

[4.5 Расчет ориентировочной цены веб-приложения 4](#_Toc72366549)9

[4.6 Расчёт основных показателей экономической эффективности …………..50](#_Toc72366549)

[Вывод 51](#_Toc72366550)

[5 Безопасность жизнедеятельности](#_Toc72366551) 53

[5.1 Анализ условий труда в офисе учебной части РФМШ, расчет…53 производственного освещения 5](#_Toc72366552)3

[5.2 Расчет естественного освещения………………………………………..….](#_Toc72366554)54

[5.3 Проверка искусственного освещения 5](#_Toc72366555)6

[5.4 Расчет освещения методом использования светового потока………..…..5](#_Toc72366556)8

[5.5 Эргономические требования к рабочему месту за ПК. 5](#_Toc72366557)9

[Вывод 61](#_Toc72366561)

[Заключение](#_Toc72366562) 62

[Список использованной литературы 6](#_Toc72366563)3

[Приложение А 6](#_Toc72366564)5

[Приложение Б](#_Toc72366564) 66

# Введение

Одной из важнейших проблем качественной организации учебного процесса в школах является задача создания автоматизированного учебного расписания. Правильно и точно составленное расписание обеспечивает равномерную нагрузку групп каждого из классов и преподавательского состава.

В настоящее время использование информационных систем в образовательных учреждениях не является редкостью. Спектр их применения широк и варьируется от автоматизации отдельно взятых рабочих мест до полной автоматизации деятельности школ.

Вне зависимости от объекта автоматизации, будь то преподавательский состав или администрация, в образовательном учреждении такие системы внедряют, преследуя конечную цель – повышение качества образования.

Школа, как и любое предприятие, непременно проходит процесс автоматизации и, несмотря на то что понятие образовательной деятельности едино для всех образовательных учреждений, в каждом школе этот процесс проходит по-разному. Значительное влияние на процессы автоматизации оказывает как наличие денежных средств, так и готовность использования предлагаемых рынком информационных услуг программных продуктов.

В связи с функционированием школ в рамках единого информационного пространства, использование сторонних программных продуктов делается невозможным ввиду специфики работы имеющихся систем или же по причине дороговизны внедрения, влекущего значительную доработку как имеющихся, так и приобретаемых информационных систем.

С целью автоматизации планирования было разработано решение, упрощающее процесс создания электронного расписания на основе анализа имеющихся учебных планов специальностей, позволяющий анализировать структуру нагрузки, а также планировать структурную доработку и некоторую унификацию имеющихся учебных планов.

Общие требования, предъявляемые к автоматизированной системе, следующие:

- использование норм времени для расчета объемов учебной нагрузки;

- использование информации из учебных планов каждой параллели;

- быстрота обработки информации за счет автоматизации возможных операций пользователя системы;

- формирование отчетных форм;

- расширяемость системы (возможность её доработки в случае повышения требований к автоматизированной системе);

- удобный и дружественный пользовательский интерфейс.

Для создания системы использовались языки программирования C#, в качестве системы управления базами данных использовалась СУБД MySQL.

# 1 Аналитическая часть

# 1.1 Анализ деятельности организации

Некоммерческое акционерное общество «Республиканская физико-математическая школа» в городе Алматы является специализированным заведением для одаренных детей по всему Казахстану и расположена по адресу Бостандыкский район, бульвар Бухар Жырау 36.

Главной целью РФМШ является выявление и дальнейшее развитие одаренных детей из всех регионов Казахстана, формирование интеллектуальной элиты страны. Многие выпускники стали влиятельными политиками, перспективными учеными и преуспевающими бизнесменами, новаторами в своей области, отличающимися особой корпоративной культурой.

РФМШ в городе Алматы обеспечивает получение образования с 1 по 11 классы включительно.

Образовательная деятельность РФМШ построена на основе принципов:

* принцип целостности образования, основанный на представлении об единстве процессов развития, обучения и воспитания;
* принцип непрерывности и преемственности образования, в рамках которого образование рассматривается как постоянный процесс на протяжении всей жизни с опорой на предыдущий опыт и ориентацией на прогнозируемый результат;
* принцип интеграции, который предполагает взаимосвязь всех компонентов процесса обучения, всех элементов системы и связь между системами;
* принцип многоуровневости, который предполагает образование на нескольких взаимосвязанных уровнях с учетом возрастных и образовательных возможностей обучающихся, задач их воспитания и творческого развития, а также формирования готовности к продолжению образования и жизни в обществе;
* принцип дифференциации и индивидуализации, который направлен на создание условий для полного проявления и развития способностей каждого школьника;
* принцип осознанного выбора, предполагающий обладание обучающимися субъективными полномочиями в выборе цели, содержания, форм и способов образования, участия в жизнедеятельности школы и местного сообщества.

Школа работает в режиме пятидневной учебной недели. В субботу проводятся дополнительные кружки, также подготовительные занятие по олимпиадным дисциплинам. Учебные занятия в школе начинаются: в 08:00, последний урок заканчивается в 15:15.

# 1.2 Организационная структура и система управления

Исходя из целей и принципов РФМШ организационная структура управления школой построена по дивизионному типу, в которой функционируют следующие уровни:

Директор является главным административным лицом, воплощающим единоначалие и несущее персональную ответственность за все, что делается в образовательном учреждении всеми субъектами управления. На этом же уровне организационной структуры находятся высшие органы коллегиального и общественного управления, имеющие тот или иной правовой статус: совет школы, педагогический совет, профсоюзная организация и обще дошкольное родительское собрание. Субъекты управления этого уровня обеспечивают единство управляющей системы в целом, определяют стратегическое направление развития школы и всех его подразделений.

Совет школы занимается организацией содействием осуществлению самоуправленческих начал, развитию инициативы коллектива, реализации прав автономии школы в решении вопросов, способствующих организации образовательного процесса и административно-хозяйственной деятельности.

Педагогический совет предназначен для объединения усилий коллектива школы для повышения уровня воспитательно-образовательного процесса, использования в практике достижений педагогической науки и передового опыта.

Профсоюзная организация занимается защитой социально – экономических прав сотрудников заведения, добивается выполнения социальных гарантий и улучшает микроклимат в коллективе.

Обще дошкольное родительское собрание предназначено для обеспечения взаимодействия семьи и школы, что является одним из ключевых принципов организации эффективного учебного процесса.

Деятельность заместителей директора связана с определенными направлениями или подразделениями учебно-воспитательной системы согласно своему административному статусу. Этот уровень выступает звеном опосредованного руководства директора образовательной системой. Главная функция заместителей директора - согласование деятельности всех участников процесса в соответствии с заданными целями, программой и ожидаемыми результатами.

В рамках данной работы ключевой интерес представляет заместитель директора по учебно-воспитательной работе (УВР), одной из задач которого является составление расписаний занятий в НАО «РФМШ».

В структурных связях принципиальным является единство управления, со управления и самоуправления. В школе разработаны функциональные обязанности для управленцев каждого уровня управления, что обеспечивает четкость и слаженность в управлении развитием РФМШ. Административные обязанности распределены согласно Уставу и штатного расписания. Грамотное распределение функциональных обязанностей обеспечивает автономное управление каждого структурного подразделения, определяет персональную ответственность руководителей подразделений за результаты труда.

# 1.3 Цель и задачи разработки

Целью является оптимизация процессов работы составления расписания школы с помощью разработки приложения и внедрения в учебный процесс.

Задачи:

* анализ предметной области;
* моделирование производственных процессов подразделения;
* выбор средств разработки;
* проектирование базы данных;
* разработка приложения для взаимодействия с базой данных.

# 1.4 Обоснование выбора ПО

Для разработки автоматизированной системы необходимо выбрать средства, с помощью которых было бы возможно разработать программное обеспечение и обеспечить информационные потребности НАО «РФМШ» в отношении составления расписания занятий [9].

В первую очередь обосновать выбор системы управления базой данных. СУБД (система управления базами данных) – это информационная среда для создания баз данных и управления хранимыми в них сведениями об объектах реального мира.

Выбор СУБД определяется многими факторами, но главный из них – возможность работы с построенной моделью данных. Поэтому одной из важнейших характеристик является тип модели (иерархический, сетевой, реляционный), который поддерживается СУБД. База данных для автоматизированной системы составления расписания в школе должна быть реляционной. Также необходимо обратить внимание на минимальные технические требования для оперативной и эффективной работы СУБД [10].

Для сравнения выберем такие СУБД, как: MS Access, MySQL, Oracle и MS SQL Server. Сравнительный анализ СУБД представлен в Приложении В. В качестве инструмента создания базы данных была выбрана СУБД MS SQL Server 2014 [11].

На сегодняшний день MS SQL Server – наиболее популярная СУБД, используемая для разработки настольных баз данных (БД). MS SQL Server успешно применяется для создания многопользовательских приложений, где файлы базы данных являются разделяемыми ресурсами в сети. Кроме того, в MS SQL Server реализована надёжная система защиты от несанкционированного доступа к файлам. Востребованность СУБД MS SQL Server обусловлена следующими причинами:

1. система полностью русифицирована;
2. благодаря идеологии Windows можно представить информацию наглядно и красочно;
3. широкое и наглядное представление справочной системы;
4. работает на различных платформах;
5. полностью многопоточная с использованием потоков ядра. Это означает, что, если такая возможность обеспечивается, можно легко организовать работу с несколькими процессорами;
6. хеш-таблицы в памяти, используемые как временные таблицы;

ж) SQL-функции реализованы при помощи хорошо оптимизированной библиотеки классов, поэтому они выполняются настолько быстро, насколько это возможно;

з) большое количество типов данных;

и) MS SQL Server может управлять очень большими базами данных.

Таким образом, использование данного программного обеспечения обусловлено наличием множества интегрированных служб, которые помогают расширить возможности применения разнообразной информации, например, осуществить поиск, выполнять синхронизацию и анализ, составлять запросы.

Для выбора средства разработки автоматизированной системы составления расписания НАО «РФМШ» проведем сравнительный анализ следующих сред разработки:

1. Microsoft Visual Studio;
2. C++ Builder;
3. Borland Delphi.

Microsoft Visual Studio – линейка продуктов компании Майкрософт, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework и Microsoft Silverlight [12].

C++ Builder – программный продукт, инструмент быстрой разработки приложений, интегрированная среда программирования, система, используемая программистами для разработки программного обеспечения на языке C++.

C++ Builder объединяет в себе комплекс объектных библиотек, компилятор, отладчик, редактор кода и многие другие компоненты. Цикл разработки аналогичен Delphi. Большинство компонентов, разработанных в Delphi, можно использовать и в C++ Builder без модификации, но обратное утверждение не верно.

Borland Delphi можно определить как комбинацию нескольких важнейших технологий:

* высокопроизводительный компилятор в машинный код;
* объектно-ориентированная модель компонент;
* визуальное (а, следовательно, и скоростное) построение приложений из программных прототипов;
* масштабируемые средства для построения баз данных.

Сравним характеристики языков программирования, используемых в описанных выше средах разработки.

Таблица 1 − Сравнение характеристик языков программирования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | С | С++ | С# | Delphi |
| Функциональный | - | +/- | +/- | +/- |
| Обобщенное программирование | - | + | + | + |
| Возможность компиляции | + | + | + | + |
| Интерпретатор командной строки | -/+ | +/- | - | - |
| Ручное управления памятью | + | + | + | + |
| Поддержка try/catch | - | + | + | + |
| Алгебраические типы данных | - | - | - | -/+ |
| Многомерные массивы | + | + | + | + |
| Целые числа с контролем границ | - | - | - | + |
| Интерфейсы | - | + | + | + |
| Макросы | -/+ | -/+ | - | - |

В качестве средства разработки был выбран программный продукт Visual Studio 2017. Использование Microsoft Visual Studio обусловлено удобством при написании кода, мощным функционалом для отладки, большим количество дополнительных компонентов, упрощающих процесс написания ПО. Среда разработки Visual Studio 2017 способна предоставить значительные возможности разработчику и имеет ряд преимуществ перед другими средами разработки, таких как:

* значительные графические возможности;
* большой выбор компонентов в свободном доступе;
* простота использования.

В то же время Microsoft Visual Studio является популярной средой разработки и используется во многих компаниях, что также послужило причиной выбора данной среды разработки.

В качестве средства реализации автоматизированной системы составления расписания школы был выбран язык программирования C# [13].

C# является объектно–ориентированным и в этом плане много перенял у Java и С++. Использование C# в целях разработки автоматизированной системы обусловлен тремя важными характеристиками: простотой, эффективностью и гибкостью.

# 1.5 Спецификация и обоснование нефункциональных требований

Перед проектированием автоматизированной системы составления расписания занятий необходимо определиться с предъявляемыми к ней нефункциональными требованиями. Можно выделить следующие группы требований к:

* программно-технической среде;
* информации, циркулирующей в автоматизированной системе;
* квалификации персонала;
* надежности;
* обеспечению информационной безопасности;
* сохранности информации при авариях;
* эргономике и технической эстетике.

Учитывая выбор модели архитектуры с автономным персональным компьютером, предоставление отдельной рабочей станции не требуется. В части технической среды дополнительной модернизации также не требуется, т.к. каждая рабочая станция учебного заведения представляет собой персональный компьютер со следующими минимально необходимыми параметрами:

* + - процессор Intel CORE i5 и более;
    - 4 Гб RAM или более;
    - SSD 250 Гб и более;
    - видеокарта с объемом памяти 512 МБ;
    - монитор с разрешением не менее 1600\*1200 px;
    - Windows 7 или более поздние версии;

Входные данные, по возможности, имеют свободный формат, но при этом осуществляется проверка на корректность вводимых символов и команд. В случае некорректного ввода, выводится сообщение об ошибке, и просьба проверить вводимые данные.

Необходимо также отметить, что автоматизированная система должна обеспечивать возможность исторического хранения данных с глубиной не менее 1 года.

Персонал должен быть обучен работе в соответствии с функциями, выполняемыми в автоматизированной системе в соответствии с предоставленной ролью. Разрабатываемая автоматизированная система будет доступна одной категории пользователей. В таблице 2 приведена роль пользователей автоматизированной системы составления расписания занятий.

Таблица 2 − Роли пользователей автоматизированной системы

|  |  |  |
| --- | --- | --- |
| Должность пользователя | Роль пользователя | Задачи пользователя и доступность вкладок |
| Заместитель директора по УВР | Специалист | Доступен полный функционал. Данная роль позволяет реализовывать информационный процесс составления расписания занятий |

Устройство хранения данных должно быть защищено от внешних физических воздействий. Специализированного обслуживания технических средств системы не требуется.

Требования к обеспечению информационной безопасности можно определить следующим перечнем:

1. целостность – информация, на основе которой принимаются решения, должна быть достоверной и точной, защищенной от возможных непреднамеренных и злоумышленных искажений;
2. доступность (готовность) – информация и соответствующие автоматизированные службы должны быть доступны, готовы к работе всегда, когда в них возникает необходимость;
3. конфиденциальность – информация должна быть доступна только тому, кому она предназначена.

Меры по обеспечению безопасности в автоматизированной системе составления расписания занятий РФМШ должны быть направлены на нейтрализацию актуальных угроз безопасности данных, связанных с получением несанкционированного доступа к информации.

Требования по сохранности информации включают перечень событий, при которых должна быть обеспечена сохранность информации в автоматизированной системе:

1. ошибки персонала в процессе эксплуатации информационной системы;
2. искажения в каналах телекоммуникации информации, поступающей от внешних источников и передаваемой потребителям, а также недопустимые изменения характеристик потоков информации;
3. сбои и отказы аппаратуры;
4. изменения конфигурации автоматизированной системы за пределы, проверенные при испытаниях.

Разрабатываемая автоматизированной системы составления расписания занятий должна иметь удобный и простой интерфейс. Поля для ввода и отображения информации должны иметь четкое, понятное название, и расположены оптимальным образом для удобства работы с системой.

Требования, предъявляемые к эргономике и технической эстетике автоматизированной системы составления расписания занятий, можно сформулировать следующим образом.

1. удобство использования системы (user–friendly);
2. наглядное отображение выходных данных и подсказок при возникновении ошибок.

# 2 Проектная часть

# 2.1 Функциональная структура

На рисунке 1 представлена функциональная модель автоматизированной системы составления расписания занятий в НАО «РФМШ».

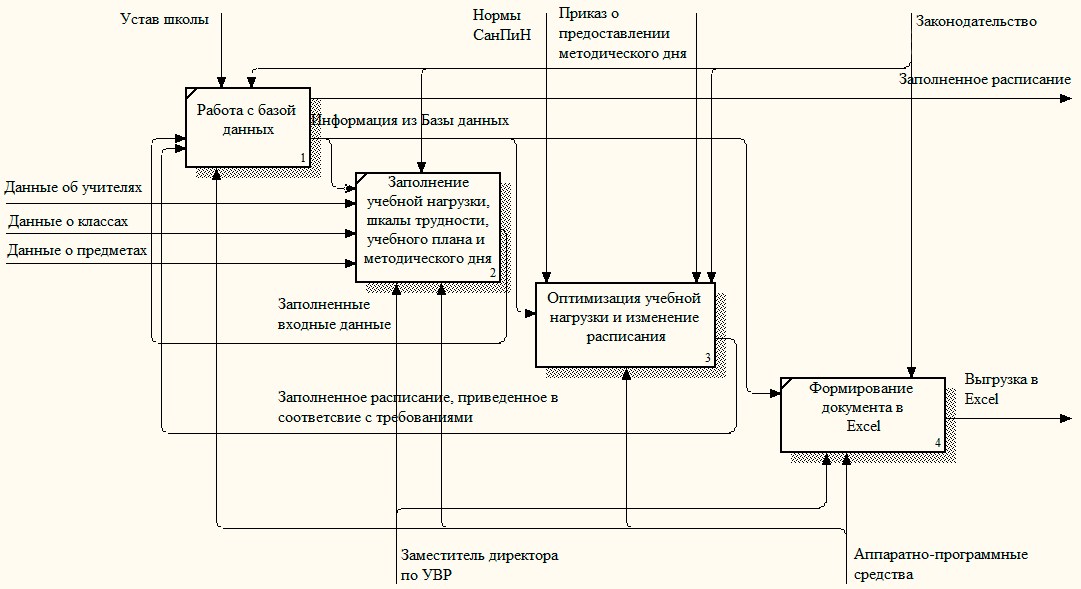


Рисунок 1 – Функциональная модель автоматизированной системы составления расписания занятий

Автоматизированная система составления расписания занятий состоит из четырех функциональных модулей: работа с базой данных, заполнение расписания, оптимизация учебной нагрузки и изменение расписания, а также формирование документа в Excel.

Модуль работы с базой данных предусматривает ввод, хранение, редактирование и удаление данных об учителях, методических днях, учебных предметах, ранге трудности каждого из предметов, числе учебных часов в неделю для каждого учебного класса, а также уровне нагрузки для каждого учебного класса. Результатом работы является база данных, которая используется во всех отдельных модулях.

Модуль заполнения расписания использует базу данных для составления расписания, учитывая требования Законодательства Казахстана. При этом на выходе мы получаем сформированные расписание занятий без учета требований СанПиН и приказа о предоставлении методических дней для учителей. Такое сформированное расписание также заносится в базу данных для дальнейшего изменения и оптимизации.

Модуль оптимизации учебной нагрузки и изменения расписания использует базу данных с целью последующей обработки информации и учета всех требований соответствующих нормативных документов. В результате на выходе формируется расписание занятий, соответствующее требованиям СанПиН и приказу о предоставлении методических дней для учителей. Такое расписание также заносятся в базу данных и используются другими модулями.

Модуль формирования документа в Excel использует базу данных с целью предоставления наглядной версии расписания для специалиста.

# 2.2 Информационное обеспечение

Инфологическая модель - это описание, которое выполнено без ориентации на используемые в дальнейшем программные и технические средства, и содержит исходную информацию о предметной области [16,17].

В инфологическом моделировании наиболее распространенной моделью является модель «сущность-связь». Именно ее следует использовать для моделирования базы данных автоматизированной системы составления расписания занятий учебного учреждения.

База данных автоматизированной системы составления расписания занятий РФМШ должна выполнять следующие функции:

1. хранение, изменение и предоставление информации об учебных предметах.
2. хранение, изменение и предоставление информации об учителях.
3. хранение, изменение и предоставление информации об учебных классах.
4. хранение, изменение и предоставление информации об учебной нагрузке.
5. хранение, изменение и предоставление информации о сформированном расписании.

Сформируем и проанализируем сущности базы данных автоматизированной системы составления расписания занятий НАО «РФМШ» в таблице 3.

Таблица 3 − Сущности и атрибуты базы данных автоматизированной системы составления расписания занятий школы

|  |  |  |
| --- | --- | --- |
| Сущность | Назначение | Атрибуты |
| Учебные предметы | Необходима для хранения и обработки сведений об учебных предметах, изучаемых в различных учебных классах | ID учебного предмета, название предмета |
| Учителя | Необходима для хранения и обработки сведений об учителях | ID учителя, ФИО учителя, методический день |

*Продолжение таблицы 3*

|  |  |  |  |
| --- | --- | --- | --- |
| Номер класса | Порядковый номер учебного класса | ID класса, номер класса | |
| Учебные классы | Необходима для хранения и обработки сведений о классах и рангах трудности, присеваемых каждому из учебных предметов для каждого класса | | ID учебного класса, ранг трудности, число учебных часов |
| Учебная нагрузка | Необходима для хранения и обработки сведений о дневной учебной нагрузке для каждого класса | | ID учебной нагрузки, значение трудности предметов |
| Расписание | Необходима для хранения и обработки сведений о расписании занятий и фактической нагрузки за день | | ID расписания, значение дневной нагрузки |

На этапе логического проектирования базы данных необходимо рассмотреть каждую связь между сущностями. Одно из основных требований к организации базы данных – это обеспечение возможности определения одних сущностей по значениям других, для чего необходимо установить между ними определенные связи. Установим в проектируемой базе данных связи посредством использования модели «сущность-связь».

Связь – это некоторая ассоциация между двумя сущностями. Связи позволяют по одной сущности находить другие, связанные с нею. Каждая связь имеет два конца. Выберем ключевые связи по разработанным сущностям и отобразим их в таблице 4.

Таблица 4 − Взаимосвязь сущностей модели базы данных автоматизированной системы составления расписания занятий

|  |  |  |
| --- | --- | --- |
| Сущность 1 | Сущность 2 | Описание |
| Учебные предметы | Учителя | Для каждого учебного предмета существует только один учитель, а каждый учитель преподает только один предмет (за исключением начальных классов). |
| Учебные предметы | Расписание | Для каждого учебного предмета определено одно расписание. В тоже время расписание содержит несколько предметов. |
| Учебные предметы | Учебные классы | Каждый учебный предмет может преподаваться в нескольких классах, в то время как в одном классе за день преподается несколько предметов |

*Продолжение таблицы 4*

|  |  |  |
| --- | --- | --- |
| Учебная нагрузка | Учебные классы | Каждая учебная нагрузка включает сумму трудности предметов для одного класса |
| Расписание | Учебные классы | Каждое расписание содержит множество учебных классов, а каждый класс отражен в расписании только один раз. |
| Расписание | Учебная нагрузка | Каждое расписание содержит множество учебных нагрузок, а каждая учебная нагрузка отражена в расписании только один раз. |

Разработаем логическую модель базы данных информационной системы и представим модель на рисунке 2.

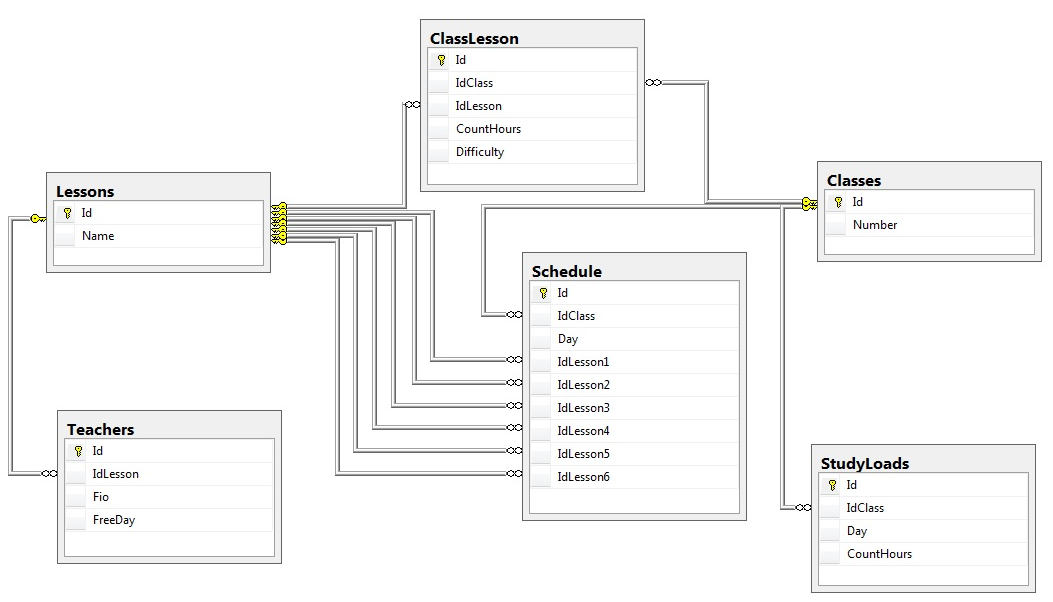


Рисунок 2 – Логическая модель базы данных автоматизированной системы составления расписания занятий

Данная модель не только отражает структуру данных, циркулирующих в автоматизированной системе составления расписания занятий, но и позволяет наглядно конкретизировать сущности, атрибуты и связи.

С целью дальнейшего проектирования базы данных разработаем физическую модель БД, содержанием которой является определение модели данных. Модель данных — это набор соглашений по способам представления сущностей, связей, агрегатов, системы классификации [17]. Каждая модель данных определяет особенности выполнения основных операций над данными:

* добавление;
* удаление;
* модификация;
* выборка.

Результатом физического проектирования является описание логической структуры базы данных на языке описания. На этапе разработки физической модели определяется состав базы данных, например, принимается решение о хранении только исходных данных, а все производные могут быть получены расчетным путем в результате запроса.

Таким образом, особое внимание при построении модели уделяют целостности и отсутствию избыточности данных. Избыточность — это многократное повторение одних и тех же данных.

Необходимо также отметить, что выходные экранные формы должны содержать информацию об учебной нагрузке, о расписании учебных предметов в разрезе классов и дней недели и соответствовать рисунку 3.

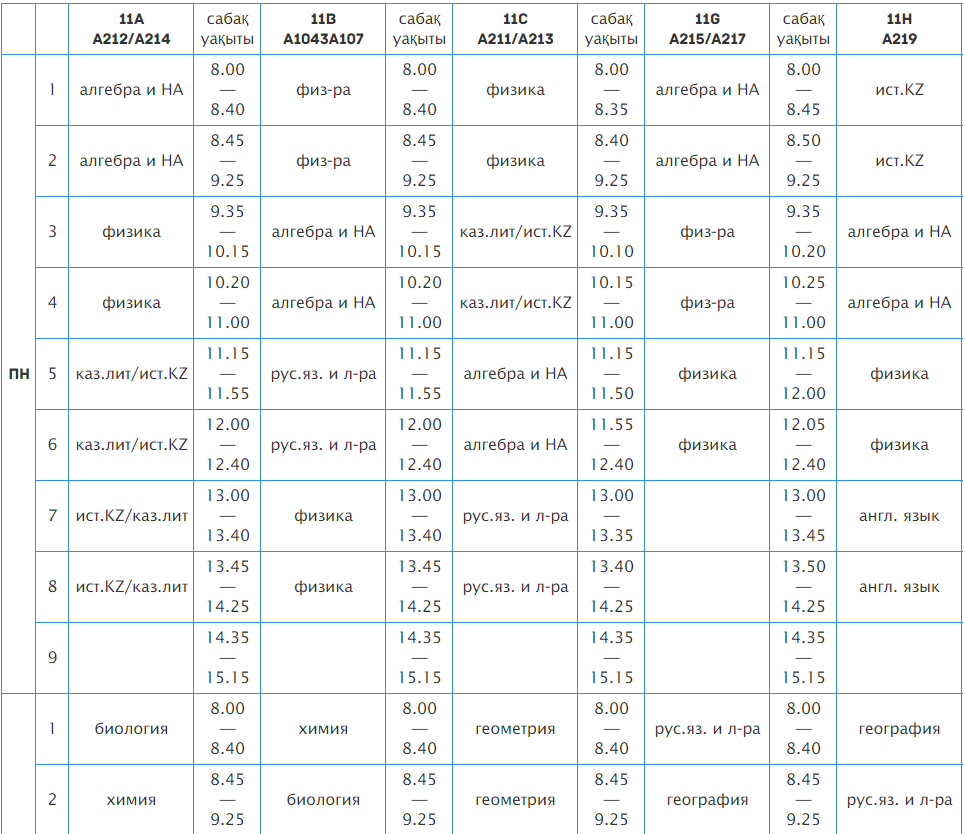


Рисунок 3 – Пример экранной формы расписания уроков НАО «РФМШ», взятый с сайта школы fizmat.kz

Первичные документы и нормативно-справочная информация, необходимые для работы автоматизированной системы составления расписания занятий, были подробно представлены в разделе 1 данной работы.

# 2.3 Математическое обеспечение

Математическую модель можно определить, как приближенное описание какого-либо класса явлений или объектов реального мира на языке математики. Целью такого моделирования является проведение исследований объектов и предсказание результатов будущих наблюдений. Как правило, математическое моделирование применяется в тех случаях, когда натурный эксперимент невозможен или затруднен по тем или иным причинам.

В общем виде, процесс математического моделирования можно условно подразделить на три этапа:

1. анализ теоретических закономерностей, свойственных изучаемому явлению или процессу, и эмпирических данных о его структуре и особенностях;
2. определение методов, с помощью которых можно построить модель;
3. анализ полученных результатов.

В рамках проектирования автоматизированной системы составления расписания занятий предложен подход к оптимизации расписания занятий в соответствие с требованиями СанПиН.

На этапе оптимизации программное обеспечение будет производить суммирование ранга трудности по всем предметам для каждого класса и календарного дня

Где, i – день недели;

j – номер класса;

k – учебный предмет

n – число предметов для каждого класса и дня недели;

– суммарная нагрузка за день;

– ранг трудности предмета для каждого класса.

В случае, если , программное обеспечение перераспределяет нагрузку, оценив уровень превышения () и выявив возможность перемещения лишнего предмета на другой день недели. Описанный алгоритм с учетом математических операций представлен на рисунке 4.



Рисунок 4 – Оптимизация расписания занятий в соответствие с нормами СанПиН

# 2.4 Программная часть

С целью проведения объектно-ориентированного проектирования автоматизированной системы составления расписания была использована нотация UML. Язык UML представляет собой графический язык моделирования общего назначения, предназначенный для спецификации, визуализации, проектирования и документирования всех артефактов, создаваемых при разработке программных систем. При проектировании программного продукта были сформированы следующие UML-диаграммы средствами Visual Paradigm for UML:

* диаграмма вариантов использования;
* диаграмма классов;
* диаграмма последовательности;
* диаграмма состояний;

В первую очередь необходимо спроектировать диаграмму вариантов использования, отражающую отношения между актёрами и прецедентами. Прецеденты – это технология определения функциональных требований к системе. Работа прецедентов заключается в описании типичных взаимодействий между пользователями системы и самой системой и предоставлении описания процесса ее функционирования.

На рисунке 5 представлена диаграмма вариантов использования. Для получения сформированного расписания и выгрузки в Excel специалисту необходимо заполнить или изменить (при необходимости) данные в справочниках «Учебные предметы», «Учителя», «Учебные классы» и «Учебная нагрузка». Работа со справочниками и сформированное расписание включены в «Составление расписания», в свою очередь «Печать расписания» – в «Сформированное расписание».

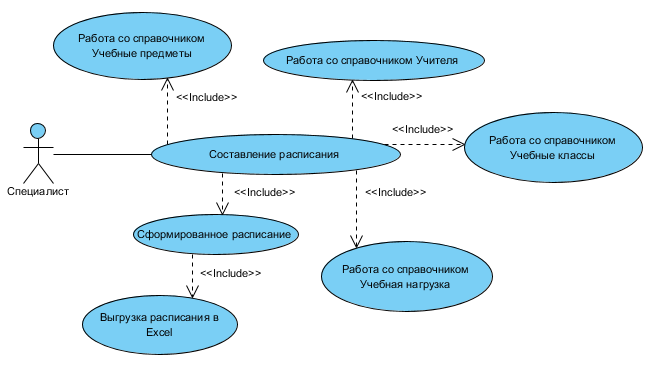


Рисунок 5 - Диаграмма вариантов использования (Use Case diagram)

Диаграммы классов являются одной из форм статического описания системы с точки зрения ее проектирования, показывая ее структуру. Диаграмма классов не отображает динамическое поведение объектов, изображенных на ней классов. На диаграммах классов показываются классы, интерфейсы и отношения между ними. Данная диаграмма представлена на рисунке 6.

Отношение обобщения представлено на диаграмме в виде зависимости классов: Главная Форма, Учебные классы, Учебные предметы, Учителя, Расписание от абстрактного класса Form. Все классы зависят от «ConnectToDB».

Проектирование осуществлялось с подходом MVC. То есть формы (view) отдельно от моделей, через которые, собственно, и происходит добавление в базу данных.

Диаграмма последовательности отражает временные особенности передачи сообщений между частями программы, а также показывает время жизни классов, которые представлены в диаграмме классов.

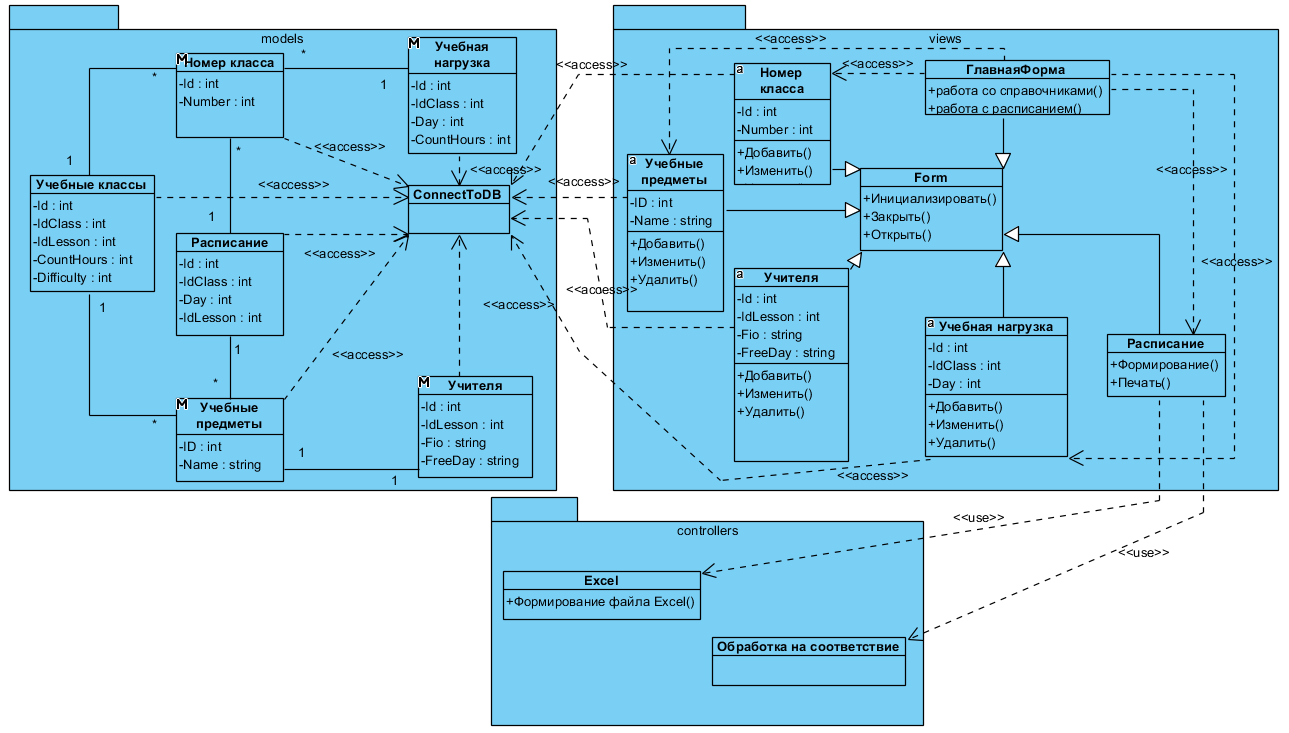


Рисунок 6 - Диаграмма классов (Class diagram)

Одним из основных принципов объектно-ориентированного программирования является способ информационного обмена между элементами системы, выражающийся в отправке и получении сообщений друг от друга. Таким образом, основные понятия диаграммы последовательности связаны с понятием объект и сообщение. Диаграмма последовательности представлена на рисунке 7.

На диаграмме показано, что после запуска программы специалист последовательно приступает к работе со справочниками «Учебные предметы», «Учителя», «Учебные классы» и «Учебная нагрузка». После заполнения всех необходимых справочников переходит к составлению расписания, при формировании которого осуществляются запросы к требуемым справочникам. По завершении работы формируется запрос на выгрузку расписания в Excel.

Диаграмма состояний показывает работу системы в динамике, характеризует жизненный цикл, который начинается при её запуске. Диаграмма представляет собой состояния, соединенные переходами.

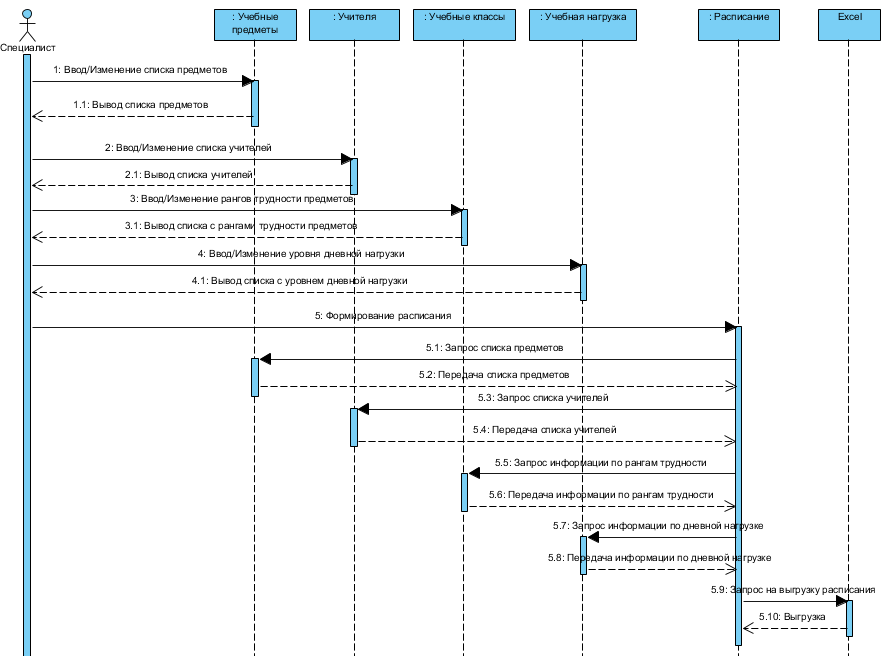


Рисунок 7 - Диаграмма последовательности (Sequence diagram)

При открытии системы отображается главное окно программы с меню. Из меню можно перейти к работе со справочниками, где для каждого свой пункт. При этом открываются соответствующие формы. Из форм справочников можно перейти на формы изменения соответствующих справочников.

Также на главной форме будут представлены кнопки, позволяющие сформировать расписание и экспортировать его в Excel.

При закрытии главного окна происходит закрытие системы. Диаграмма состояний программы показана на рисунке 8.

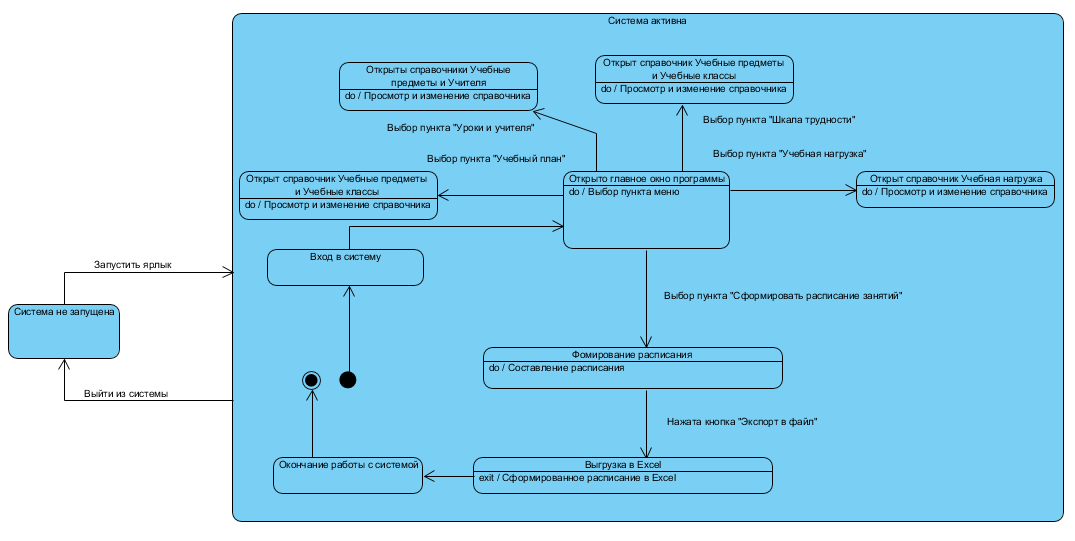


Рисунок 8 – Диаграмма состояний (Activity diagram)

Таким образом, автоматизированная система составления расписания занятий состоит из следующих модулей: cправочная информация, формирование расписания и выгрузка в Excel. Структура программного обеспечения представлена на рисунке 9, а описание модулей представлено в таблице 5.

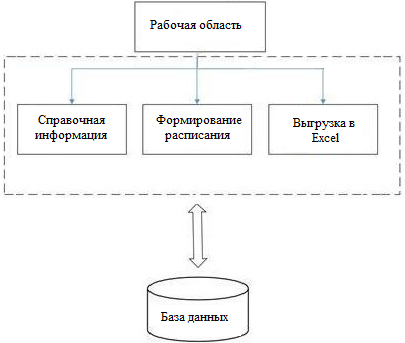


Рисунок 9 – Структура автоматизированной системы составления расписания занятий

Таблица 5 − Описание модулей автоматизированной системы составления расписания занятий

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Функциональное назначение | Входные данные | Выходные данные |
| Рабочая область | Навигация | Учебные предметы, Учителя, Учебные классы, Учебная нагрузка | Справочная информация, формирование расписания, выгрузка в Excel |
| Справочная информация | Заполнение справочной информации | Учебные предметы, Учителя, Учебные классы, Учебная нагрузка | Таблицы со справочной информацией |
| Формирование расписания | Формирование расписания занятий, соответствующего выявленным требованиям | Учебные предметы, Учителя, Учебные классы, Учебная нагрузка | Таблица с расписанием занятий |
| Выгрузка в Excel | Выгрузка заполненного расписания в Excel | Сформированное расписание | Расписание занятий в формате Excel |

Обобщенный алгоритм работы автоматизированной системы составления расписания занятий, включающий предложенный выше алгоритм оптимизация расписания занятий в соответствие с нормами СанПиН, представлен на рисунке 10.



Рисунок 10 - Обобщенный алгоритм работы автоматизированной системы составления расписания занятий

Опишем ключевые этапы разработки автоматизированной системы составления расписания занятий. На рисунке 11 представлена часть программного кода алгоритма по составлению расписания для 1 - 4 классов.

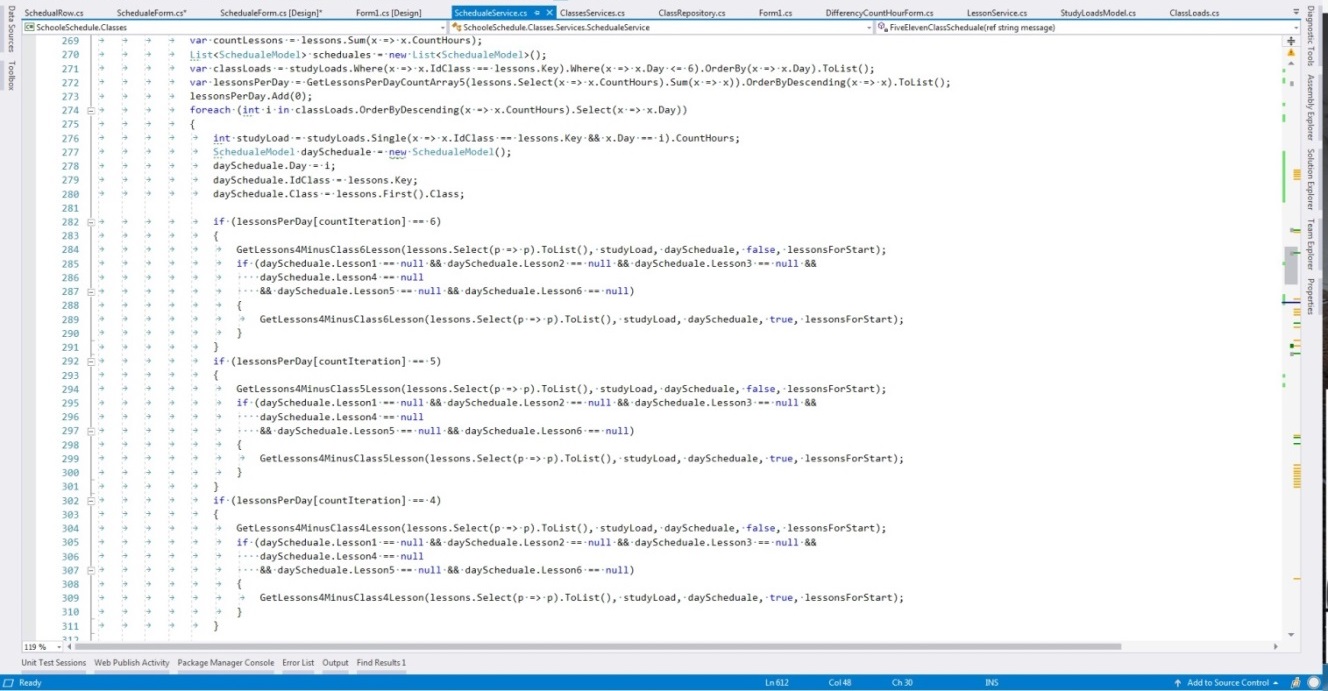


Рисунок 11 - Программный код алгоритма по составлению расписания для 1 - 4 классов

Представленный на рисунке 11 алгоритм можно охарактеризовать следующей последовательностью:

1. получаем общую коллекцию уроков;
2. рассчитываем сколько всего в неделю будет уроков у класса, после чего раскладываем это число на пять слагаемых (количество уроков на каждый день). При этом учитывается условие, что в день может быть не меньше четырех уроков;
3. для дня с самой максимальной загрузкой берем максимальное количество уроков и в функциях GetLesson4Minus ищем оптимальное коллекцию из количества уроков, сумма нагрузки по которым будет максимально близко к учебной нагрузке текущего дня. При подходящей коллекции проверяем, что между уроками нет промежутков и в день не более 2 одинаковых уроков;
4. повторяем предыдущий шаг для каждого дня каждого класса;
5. в результате получаем коллекцию из 20 дней, которая записывается в базу данных.

На рисунке 12 представлена часть кода алгоритма по составлению расписания для 7 - 11 классов.

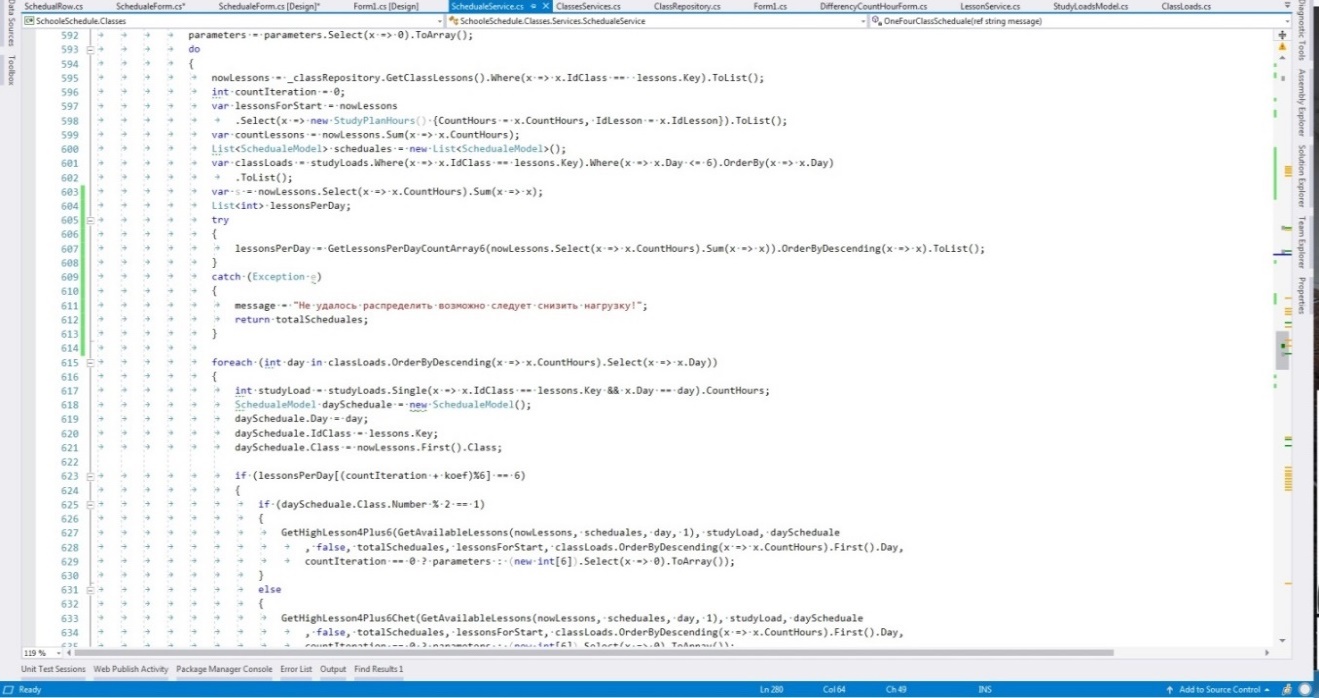


Рисунок 12 - Программный код алгоритма по составлению расписания для 7 - 11 классов

Представленный на рисунке 13 алгоритм можно охарактеризовать следующей последовательностью:

1. получаем общую коллекцию уроков;
2. рассчитываем сколько всего в неделю будет уроков у класса с самым большим количеством уроков, после чего раскладываем это число на 6 слагаемых (количество уроков на каждый день). При этом учитывается условие, что в день может быть не меньше четырех уроков;
3. для дня с самой максимальной загрузкой берем максимальное количество уроков и в функциях GetLesson4Plus ищем оптимальное коллекцию из 4 - 6 уроков, сумма нагрузки по которым будет максимально близка к учебной нагрузке текущего дня. Из общей коллекции уроков получаем коллекции, которые могут быть в этот день (отсеиваем те, у которых учителя в этот день имеют методический выходной и те уроки, которые уже записаны в расписании для любого другого класса). После чего проверяем, что у класса может быть это количество уроков в этот день и в день не более 2 одинаковых уроков;
4. повторяем предыдущий шаг для каждого дня этого класса и в конце проверяем все ли уроки мы смогли распределить. При этом осуществляется проверка на минимальное число уроков (четыре урока) и если все проверки прошли удачно, добавляем эти 6 дней в коллекцию;
5. если проверки не удачны, то возвращаемся к первому дню и начинаем дальше искать лучшую комбинацию (сумма рангов трудности предметов максимально близка к максимальной сложности этого дня);
6. в случае если все возможные комбинации алгоритм перебрал, то меняем распределения количества уроков ко дню (если в первые раз мы считали на 1 день - 6, 2 день - 5, 3 день - 4, то теперь будет 1 день - 4, 2 день - 6, 3 день - 5);
7. после перебора всех вариантов, доступных алгоритму, приходим либо к финальной коллекции из 42 дней для 7 классов (с 1 по 11) либо выводим ошибку, о том, что учебную нагрузку невозможно распределить в заданных условиях.

На рисунке 13 представлена функция, которая рассчитывает не превышена ли максимальная нагрузка для какого-либо класса, складывая нагрузку каждого урока для этого класса друг с другом.

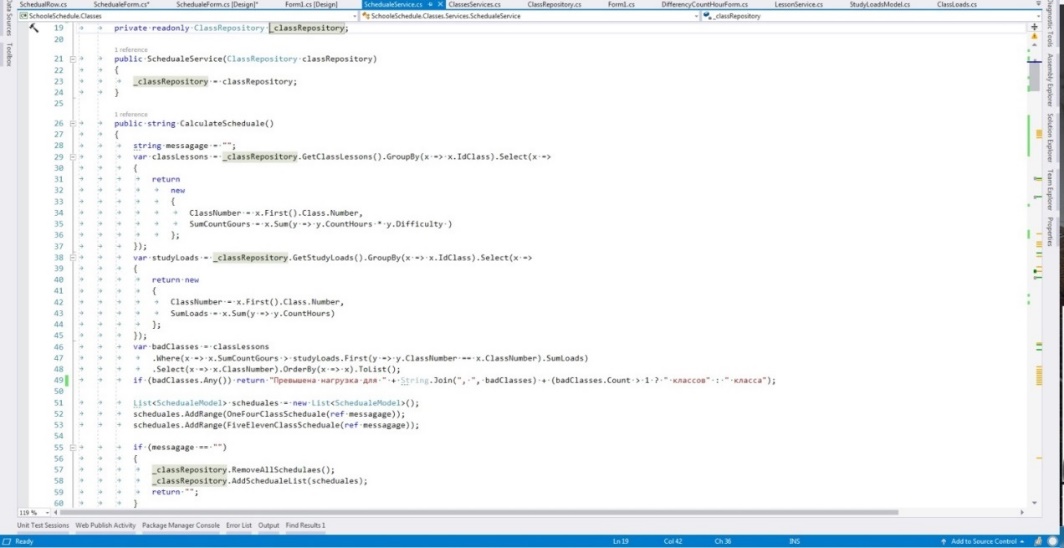


Рисунок 13 – Функция оценки превышения учебной нагрузки

На рисунке 14 показана процедура экспорта содержимого таблицы расписания в Excel файл. Пользователю необходимо выбрать файл, в который будут сохранены данные, после чего данные из всех ячеек сформированного расписания записываются в соответствующую ячейку Excel файла.

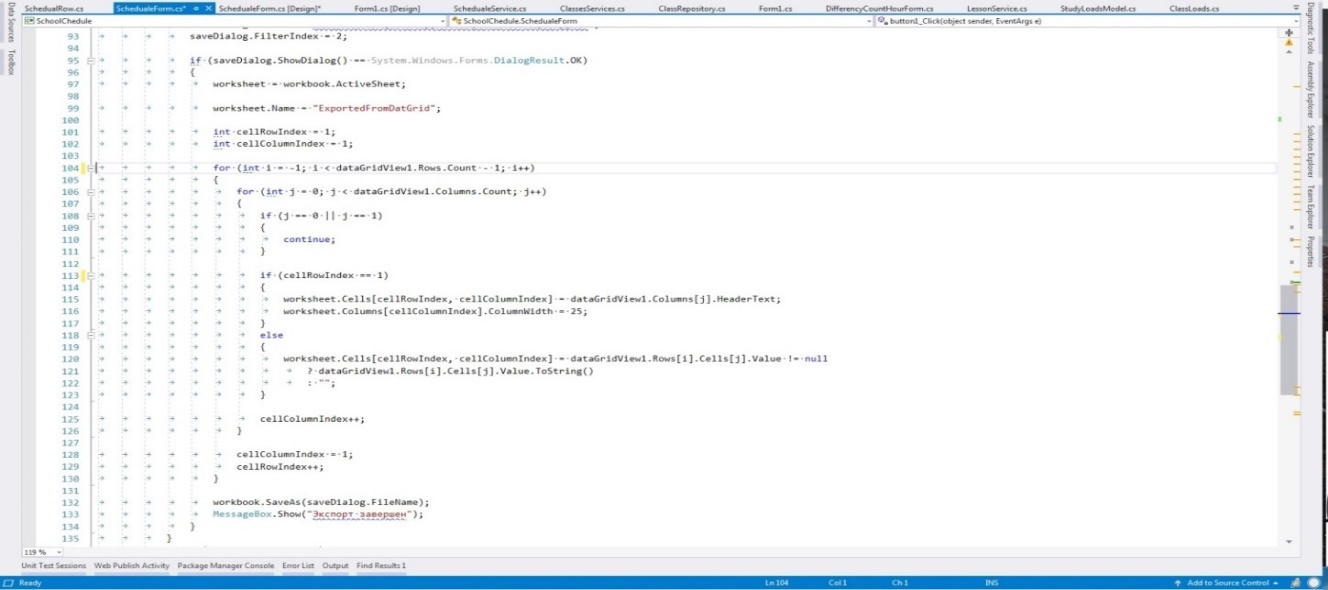


Рисунок 14 - Процедура экспорта содержимого таблицы расписания в Excel

На рисунке 15 представлена диаграмма развертывания, на которой представлены узлы выполнения программных компонентов, а также процессов и объектов.

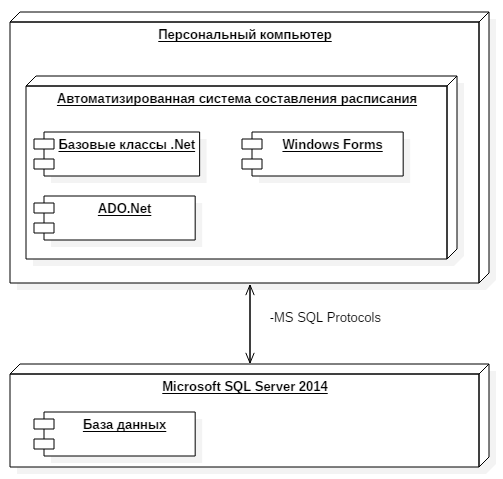


Рисунок 15 – Диаграмма развертывания

# 2.5 Обеспечение информационной безопасности

В связи с тем, что автоматизированная система составления расписания занятий будет функционировать на одной рабочей станции, а обрабатываемая информация не будет содержать персональных данных, предпринимать дополнительные организационно-технических меры по защите информации не требуется. Права доступа будут предоставлены согласно таблице 2 данной работы.

В то же время, с целью минимизации риска реализации угроз информационной безопасности в работы проведен экспертный анализ вероятности возникновения угроз и их опасности, а также определены актуальные угроз для автоматизированной системы составления расписания занятий. Результаты анализа приведены в таблице 6.

Таблица 6 − Оценка актуальности угроз информационной безопасности автоматизированной системы составления расписания занятий

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Угроза безопасности автоматизированной системы составления расписания занятий | Вероятность возникновения угрозы | Вероятность реализации угрозы | Опасность реализации угрозы | Актуальность угрозы |
| Угроза типа "Отказ в обслуживании" | Низкая | Средняя | Средняя | Актуальная |
| Угроза "Анализа сетевого трафика" с перехватом передаваемой по сети информации | Низкая | Средняя | Низкая | Неактуальная |
| Угроза незаконного получения паролей и других реквизитов разграничения доступа с дальнейшим их использованием | Низкая | Средняя | Средняя | Актуальная |
| Угроза разглашения, передачи или утраты атрибутов разграничения доступа | Низкая | Средняя | Средняя | Актуальная |
| Угроза чтения остаточной информации из оперативной памяти и с внешних запоминающих устройств | Низкая | Средняя | Низкая | Неактуальная |
| Угроза чтения информации из областей оперативной памяти, используемых операционной системой | Низкая | Средняя | Низкая | Неактуальная |
| Угроза реализации скрытого канала передачи данных | Низкая | Средняя | Средняя | Актуальная |
| Угроза незаконного подключение к линиям связи с целью прямой подмены законного пользователя путем его физического отключения после входа в систему и успешной аутентификации с последующим вводом дезинформации и навязыванием ложных сообщений | Низкая | Средняя | Низкая | Актуальная |
| Угроза внедрения программных "закладок" и "вирусов", которые представляют собой вредоносные программы | Низкая | Средняя | Средняя | Актуальная |
| Угроза внедрения аппаратных закладок | Низкая | Средняя | Низкая | Неактуальная |
| Угроза внедрения агентов в число персонала системы | Средняя (5) | Средняя | Средняя | Актуальная |
| Угроза вывода из строя подсистем обеспечения функционирования сети | Низкая | Средняя | Низкая | Неактуальная |

*Продолжение таблицы 6*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Несанкционированное отключение средств защиты | Средняя | Средняя | Средняя | Актуальная |
| Угроза модификации системных файлов и средств защиты информации в целях вывод их из строя | Низкая | Средняя | Низкая | Неактуальная |
| Непреднамеренный вывод из строя аппаратно-программных средств | Низкая | Средняя | Низкая | Неактуальная |
| Угрозы получения доступа к информации, основанные на использовании уязвимостей системного и прикладного ПО при нарушении ограничений и условий эксплуатации ПО | Низкая | Средняя | Низкая | Неактуальная |
| Установка ПО, не связанного с исполнением служебных обязанностей | Низкая | Средняя | Низкая | Неактуальная |
| Непреднамеренная модификация (уничтожение) информации сотрудниками | Низкая | Средняя | Низкая | Неактуальная |
| Непреднамеренное отключение средств защиты | Низкая | Средняя | Низкая | Неактуальная |
| Сбой системы электроснабжения | Низкая | Средняя | Низкая | Неактуальная |
| Стихийное бедствие | Низкая | Средняя | Низкая | Неактуальная |

По результату проведенного анализа можно рекомендовать меры по обеспечению безопасности для рабочей станции, на которой будет функционировать автоматизированная система составления расписания занятий. Такие меры должны быть направлены на нейтрализацию указанных таблице 6 угроз безопасности данных. Рекомендуемые программно- аппаратные средства защиты информации можно перечислить в следующем списке:

1. средство защиты информации от несанкционированного доступа (СЗИ Dallas Lock 8.0-K);
2. антивирусное средство (Dr.Web Enterprise Security Suite 11.0);
3. межсетевой экран (МЭ «Ideco UTM 7»).

Среди организационных мер по защите информации можно рекомендовать следующие:

1. организационные меры по работе со съёмными носителями информации;
2. организационные меры по периодической смене паролей (не реже одного раза в месяц);
3. организацию работы администратора безопасности.

В то же время, с целью минимизации риска реализации угроз, описанных в таблице 6, необходимо организовать резервное копирование и восстановление данных автоматизированной системы составления расписания занятий в случае сбоя.

# 3 Экспериментальная часть

# 3.1 Контрольный пример

При запуске программы открывается главная форма, представленная на рисунке 16.

По умолчанию форма содержит открытую вкладку с информацией об учебной нагрузке в разрезе классов. Меню включает в себя две кнопки «Уроки и учителя» и «Сформировать расписание занятий».

Для того чтобы изменить входные данные по рангам трудности и часам учебных занятий предназначены вкладки «Шкала трудности» и «Учебный план», представленные на рисунках 17 и 18 соответственно.

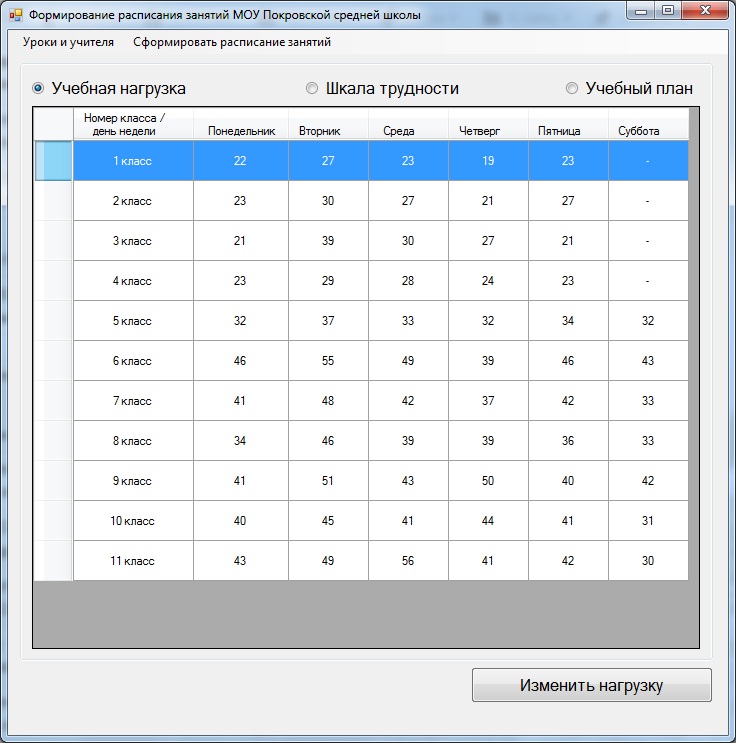


Рисунок 16 – Главная форма программы с активной вкладкой учебной нагрузки

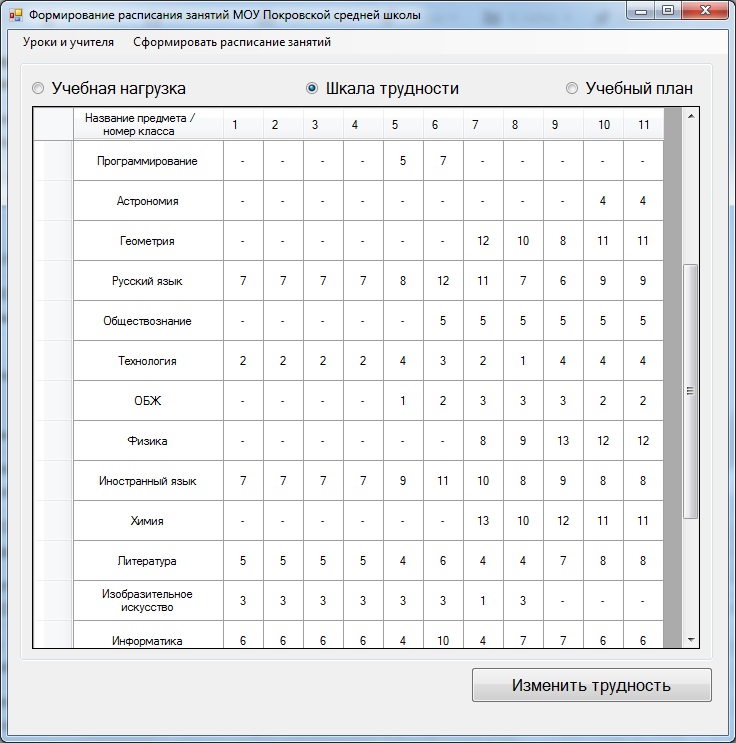


Рисунок 17 – Главная форма программы с активной вкладкой шкалы трудности

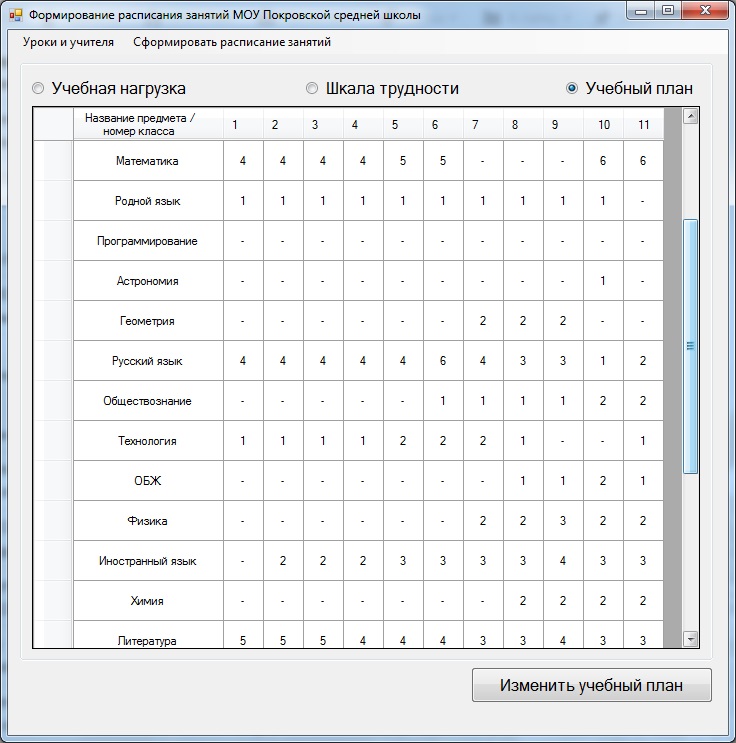


Рисунок 18 – Главная форма программы с активной вкладкой учебного плана

Для каждой вкладки предусмотрена кнопка “Изменить”, при нажатии которой появляется форма для ввода новых данных. Так, на рисунке 19 представлена форма изменения учебной нагрузки.

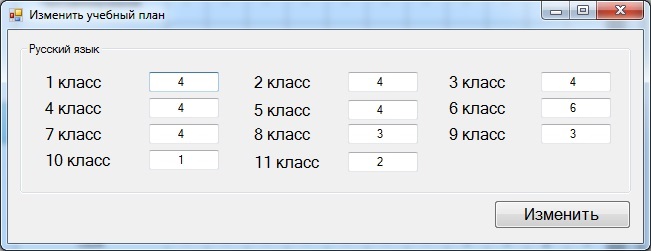


Рисунок 19 – Форма изменения учебной нагрузки

В автоматизированной системе составления расписания аналогичные формы разработаны для вкладок «Шкала трудности» и «Учебный план».

Изменение данных по урокам, учителям и методическим дням реализуется через пункт меню «Уроки и учителя», при выборе которого появляется форма (20).

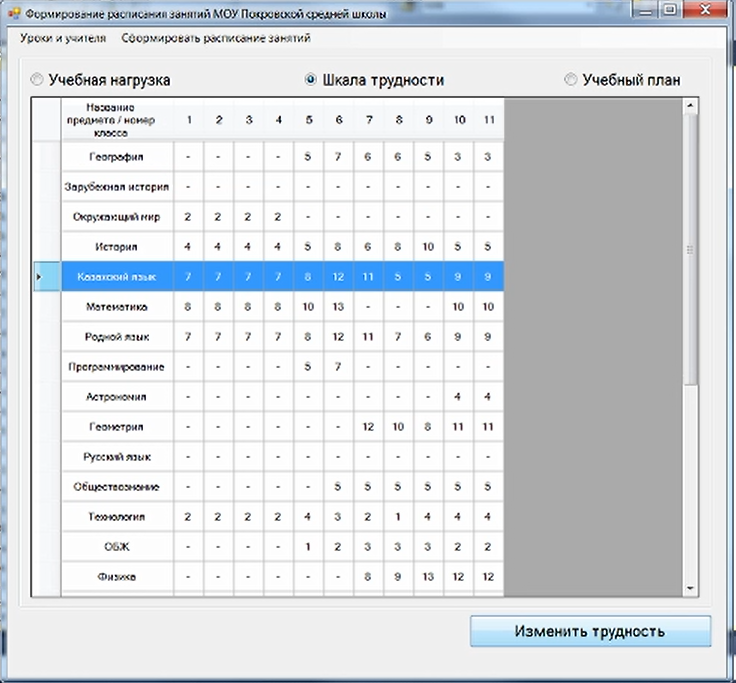


Рисунок 20 – Форма, отражающая перечень уроков, учителей и методических дней

Форма, отражающая перечень уроков, учителей и методических дней, содержит кнопки «Добавить» и «Изменить», при нажатии которых появляется возможность добавления/изменения данных посредством формы, представленной на рисунке 21.

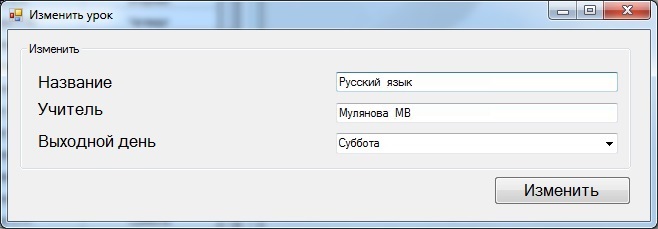


Рисунок 21 – Форма добавления / изменения данных по урокам и методическим дням

После того, как все входные данные заполнены, пользователь имеет возможность получить сформированное расписание, нажав кнопку «Сформировать расписание занятий». В результате появляется новое окно, содержащее заполненное расписание (рисунок 22).

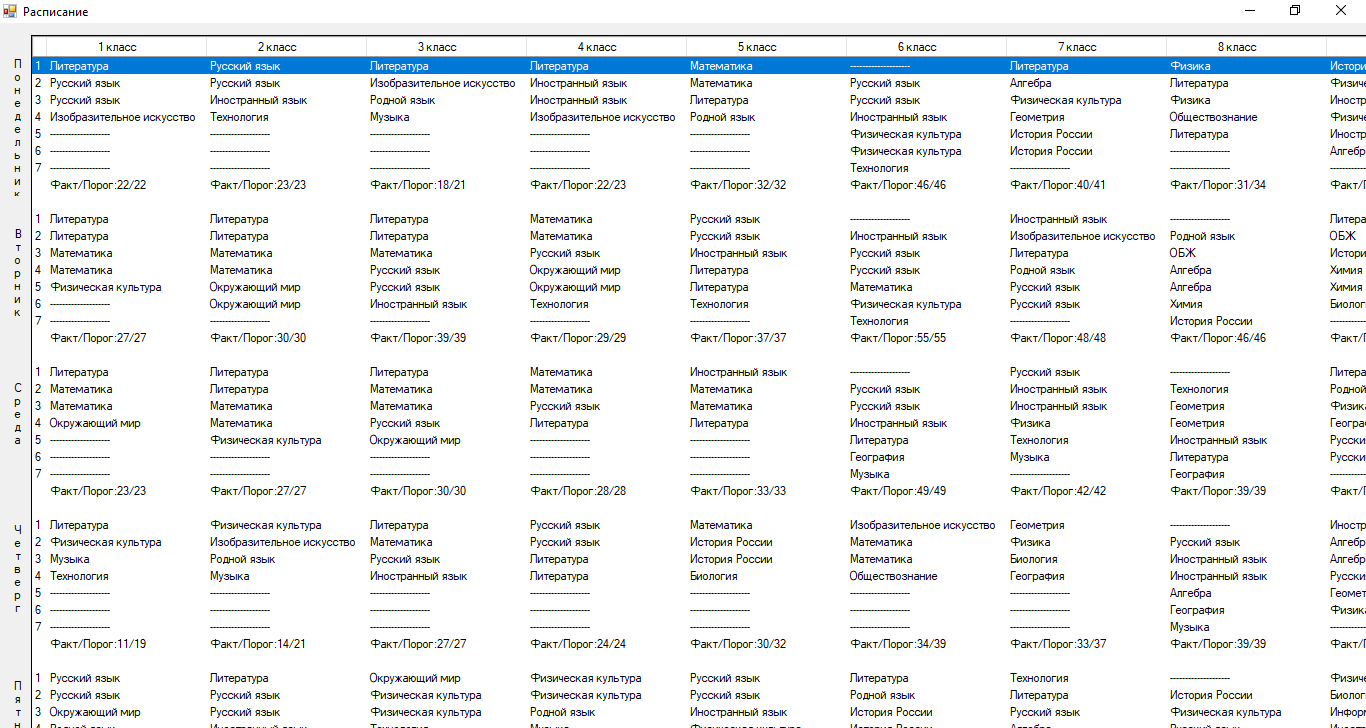


Рисунок 22 – Сформированное расписание

В случае, если необходимо выгрузить сформированное расписание, пользователь нажимает на кнопку «Экспорт в файл» и получает сохраненную информацию в Excel файле (рисунок 23).

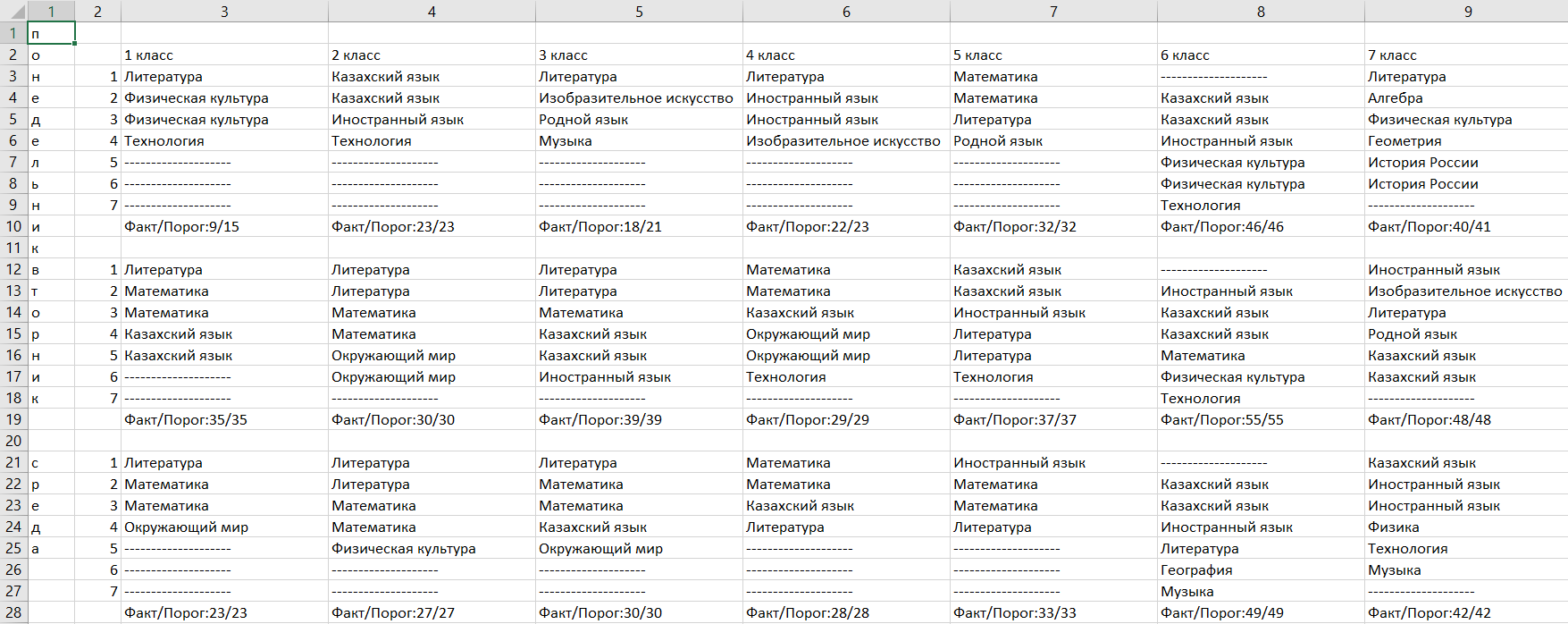
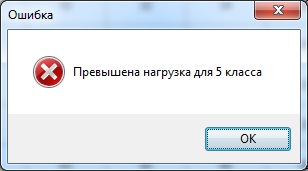
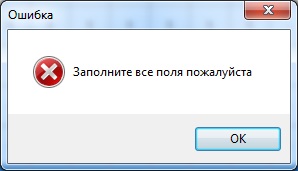


Рисунок 23 – Excel файл с выгруженным расписанием

В процессе разработки автоматизированной системы составления расписания были предусмотрены валидационные сообщения, примеры которых представлены на рисунке 25.



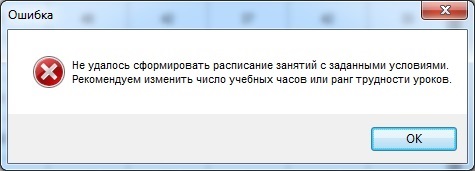


Рисунок 25 – Валидационные сообщения

Для сокращения числа ошибок пользователя в программе больше всего используются combobox, ограничения на ввод символов в textbox, а также кнопки становятся доступными только в нужном порядке. Условия появления валидационных сообщений представлены в таблице 7.

Таблица 7 − Сообщения пользователю о возникающих ошибках

|  |  |  |  |
| --- | --- | --- | --- |
| Форма | Условия появления | Текст сообщения | Реакция пользователя |
| Добавление/изменение информации по урокам и методическим дням | Поле пустое или содержит некорректные данные | Ошибка! Заполните все поля, пожалуйста | Закрыть окно, заполнить все поля корректными данными |
| Изменение учебной нагрузки | Общая сумма учебных часов превышает допустимую нагрузку | Ошибка! Превышена нагрузка для “\*” класса | Закрыть окно, изменить нагрузку для “\*” класса |
| Формирование расписания | Появляется в случае, если программа не в состоянии распределить все уроки | Ошибка! Не удалось сформировать расписание занятий с заданными условиями. Рекомендуем изменить число учебных часов или ранг трудности предметов. | Закрыть окно, изменить число учебных часов или ранг трудности предметов |

# 4 Технико-экономическое обоснование проекта

Результатом этой дипломной работы будет приложение, которое будет автоматически составлять расписание исходя из данных, которое вбил человек. Данная система уменьшает денежные расходы на премиальные людям, которые составляют расписание, а также помогает школе уменьшить затраты по времени, что уменьшает погрешность человеческого фактора.

Данная программа была смоделирована одним человеком, исходя из определенных обстоятельств, а именно: не было программ до этого в этой школе; местом прохождения моей преддипломной практики – IT-отдел.

Технико-экономическое обоснование моего дипломного проекта и программы состоит из нескольких следующих пунктов:

* определение объёма работы и трудоёмкости разработки приложения;
* расчёт затрат на разработку данного приложения;
* смета затрат на разработку данного приложения;
* расчёт возможной цены разрабатываемого приложения;
* расчёт основных показателей экономической эффективности и срока окупаемости проекта;
* выводы технико-экономической части дипломного проекта.

# 4.1 Определение объёма и трудоёмкости разработки веб-приложения

Основной частью по основным затратам является определение объема и трудоемкости разрабатываемого приложения. Процесс создания был разделен на несколько этапов и выражен в виде человек\*час показано в таблице 8.

Таблица 8 – Этапы создания приложения и оценка их в чел\*ч

|  |  |  |
| --- | --- | --- |
| Наименование этапа создания | Описание задачи | Количество в чел\*ч |
| 1 | Определение задачи проекта | 5 |
| 2 | Выбор средств разработки | 3 |
| 3 | Создание серверной части приложения | 12 |
| 4 | Создание клиентской части приложения | 67 |
| 5 | Тестирование, отладка ошибок | 10 |
| 6 | Внедрение готового продукта | 15 |
| Итого, трудоемкость разработки приложения | | 112 |

Количество рабочих дней рассчитываются следующим образом: общее количество человек\*ч делится на продолжительность рабочего дня (в нашем случае – 8 часов). Итого получается 14 рабочих дней ровно.

# 4.2 Расчёт затрат на разработку веб-приложения

При разработке приложения были задействованы средства и материалы, принадлежащие школе НАО «РФМШ» города Алматы, а именно ноутбук Asus Predator 17, принтер Canon 421, помимо этого покупалась программа Visual Studio 19, так как она лучше всего подходила для написания проектов. Все затраты приведены в таблице 9.

Таблица 9 – Затраты на разработку (специальные средства и материалы)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Наименование | Название/модель/  марка | Единица измерения | Количество | Цена за единицу, тг | Общая сумма, тг |
| Ноутбук | Asus Predator 17 | Шт. | 1 | 500000 | 500000 |
| Мышь проводная | HP office | Шт. | 1 | 2000 | 2000 |
| Принтер | Canon 421 | Шт. | 1 | 170000 | 170000 |
| Картриджи для принтера | 052 X10000 (расширенный) | Шт. | 1 | 20000 | 20000 |
| Интернет | Kaztrankom | месяц | 1 | 5000 | 5000 |
| Среда разработки | Visual Studio 19 | Шт. | 1 | 2600 | 2600 |
| Тетрадь | Обычная 48 л. | Шт. | 1 | 200 | 200 |
| Ручка | обычная, шариковая | Шт. | 1 | 200 | 200 |
| Бумага офисная | «Xerox» | упаковка | 1 | 2500 | 2500 |
| Итого | 702500 | | | | |

Так как создание приложения занимает определенное количество времени (12 рабочих дней) электроэнергия, которую мы использовали также входит в затраты для разработки и представлена в таблице 7. Время работы оборудования будет учитывать полные данные человек/час для освещения, ноутбука и принтера. Формула для вычета потраченной электроэнергии приведена на формуле 1:

(1)

где, – паспортная мощность i-го электрооборудования, кВт;

– коэффициент использования мощности i-го электрооборудования ( = 0,7..0,9);

– время работы i-го оборудования за весь период разработки ПО, ч.;

– цена электроэнергии (тариф), тг/кВт×ч.;

Таблица 10 – Затраты на электроэнергию

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Наименование оборудования/  процесса | Паспортная мощность, кВт | Коэф. испол. мощности | Время работы оборудо-вания, ч | Цена элект-роэнергии, тг/кВт·ч | Сумма, тг |
| Ноутбук | 0,8 | 0,8 | 112 | 19,00 | 1 361,92 |
| Принтер | 0,4 | 0,8 | 112 | 19,00 | 680,96 |
| Освещение | 0,3 | 0,7 | 112 | 19,00 | 446,88 |
| Итого | | | | | 2 489,76 |

Необходимо, так же, рассчитать оплату труда специалистов, для этого необходимо вычислить месячный фонд каждого сотрудника по формуле 2:

(2)

где – количество рабочих часов сотрудника за месяц;

– количество рабочих дней за месяц;

– количество рабочих часов в день.

В нашем случае, все расчеты по заработной плате специалистов будут выполняться только для одного специалиста (меня), в виду неимения других участников разработки.

Чм = 14\*8 = 112 ч

Далее, необходимо вычислить часовую ставку разработчика по формуле 3. Средняя заработная плата разработчика в Казахстане, согласно статистике за последний год, составляет 120 620 тг.

(3)

где – месячная заработная плата i-го сотрудника, тг;

– месячный фонд рабочего времени i-го сотрудника, час.

ЧСi = = 1 076,96 тг/ч

Так как в этой работе я сам предоставляю услуги разработчика и руководителя проекта, трудоемкость проекта будет рассчитываться по всем пунктам разработки:

Тразработчика = 5+3+12+67+10+15= 112 чел.\*ч

Теперь нужно определить основную заработную плату веб-разработчика по формуле 4:

(4)

где, – часовая ставка i-го работника, тг;

– трудоёмкость разработки ПО, чел.×ч.;

i – категория сотрудника;

n – количество сотрудников.

Зтр = 1076,96 \* 112 = 120619.52 тг

Для удобства восприятия, все затраты на оплату специалиста приведены в таблице 11.

Таблица 11 – Затраты на основную заработную плату труда специалиста

|  |  |  |  |
| --- | --- | --- | --- |
| Специалист | Трудоёмкость разработки ПО, чел.×ч. | Часовая ставка специалиста, тг/ч | Сумма осн. з/п, тг |
| Веб-программист | 112 | 1076,96 | 120619,52 |
| Итого | | | 120619,52 |

Для полноты отчетности данной секции, необходимо найти дополнительную заработную плату по формуле 5:

(5)

Здоп. = 120619,52 \* 0.1 = 12061,952

И в довершении, нужно вычислить фонд оплаты труда специалиста по формуле 6:

(6)

= 12061,952 + 120619,52 = 132681,47 тг

Пенсионные отчисления составляют 10% от фондов оплаты труда, получив значения которых, можно рассчитать налоги, оплаченные за проект (таблица 5).

.

Таблица 12 - Расчёт затрат на налоги

|  |  |  |  |
| --- | --- | --- | --- |
| Налог | Фонд оплаты труда (), тг | Формула для вычисления | Сумма, тг |
| СО | 132681.47 | ( – ОПВ)\*3,5% | 4179.46 |
| ОСМС | \*2% | 2653.63 |
| СН | ( – ОПВ – ОСМС)\*9,5% – СО | 6 912,71 |
| Итого | | | 13 745,8 |

# 4.3 Амортизация основных фондов (ОФ)

Основными фондами в данном проекте считаются ноутбук и принтер. Вычислим норму амортизации для ноутбука и принтера, по формуле 7:

, (7)

где ТNi– возможный срок использования i-го ОФ, год;

HАноут = = 16.67%

HАпринтер = = 10.00%

Ноутбук использовался для поиска необходимой информации и литературы, проектировании серверной и клиентской части, создании динамической БД, тестирования, поиску ошибок и неполадок приложения. Согласно таблице 1, время работы ноутбука в днях можно определить следующим образом:

Nноут = = 11 дней 1 час

Принтер же будет задействован в разработке всего 1 рабочий день, в виду ненадобности его постоянного использования. Берем в расчет, что рабочий месяц составляет 21 день и по формуле 8 находим общую сумму амортизационных отчислений.

(8)

где – стоимость i-го ОФ, тг;

– годовая норма амортизации i-го ОФ, %;

– время работы i-го ОФ за весь период разработки ПО, дней;

ЗА = + = 3 747,1 тг

Таблица 13 – Амортизация основных фондов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование оборудования | Стоимость оборудова-ния, тг | Годовая норма амортизации, % | Время работы оборудования, дни | Сумма, тг |
| Ноутбук | 290500 | 16.67 | 11,125 | 3 679,64 |
| Принтер | 256000 | 10.00 | 1 | 67,46 |
| Итого | | | | 3747,1 |

# 4.4 Смета затрат на разработку ПО

Смета затрат на разработку приложения представлена в таблице 14, а ее график в процентном распределении представлен на рисунке 26.

Таблица 14 – Смета затрат на разработку веб-приложения

|  |  |  |
| --- | --- | --- |
| Наименование статей затрат | Сумма, тг | % от общей суммы |
| Затраты на материалы и программные средства | 702500 | 83,3 |
| Затраты на электроэнергию | 2489,76 | 0.3 |
| Затраты на оплату труда специалистов | 120619.52 | 14.3 |
| Затраты на социальный налог | 13745,8 | 1.6 |
| Амортизация основных фондов | 3747,1 | 0.44 |
| Итого | 843102,18 | 100 |

Рисунок 26 – Смета затрат на разработку веб-приложения

# 4.5 Расчет ориентировочной цены веб-приложения

Для начала найдем стоимость программного обеспечения по формуле приведённой ниже (9):

(9)

где – затраты на разработку программного обеспечения, тг;

P – средний уровень рентабельности ПО, (%) (принят равным 25%).

1053877,73

Далее, нужно определить реализацию проекта с учетом НДС, ставка которого составляет 12% для ПО. Это можно сделать по формуле 10:

(10)

Полученную сумму лучше округлить, тогда получим – тг.

# 4.6 Расчёт основных показателей экономической эффективности

По подсчетам, использование приложение, которое будет автоматически составлять расписание, в месяц будет сокращать издержки, в виде премиальных выплат на 50 000 тенге, что в год составляет 600 000 тенге. Налоговые выплаты будут составлять 600 000\*10,36%=62160. Таким образом, совокупный экономический эффект будет 600 000+62 160=662 160 тенге.

Чтоб просчитать основные показатели экономической эффективности необходимо вычислить коэффициент капитальных вложений по формуле 11:

(11)

где – нормативный срок окупаемости капитальных вложений, лет. Для ПО примем равным пяти годам ().

Далее, можно вычислить ожидаемый годовой экономический эффект по формуле 12:

(12)

где – ожидаемый годовой экономический эффект, тг:

– ожидаемая условно-годовая экономия, тг;

– капитальные вложения, тг;

Ожидаемая условно-годовая экономия в данном случае является условной, так как данный параметр невозможно просчитать с данным положением дел.

Следующим действием необходимо узнать расчетный коэффициент экономической эффективности капитальных вложений по формуле 16:

(16)

где – расчётный коэффициент экономической эффективности капитальных вложений;

– ожидаемая годовая экономия, тг;

К — капитальные вложения на создание системы, тг.

Из чего мы можем узнать расчетный срок окупаемости капитальных вложений.

Из данного раздела можно привести основные показатели экономической эффективности в таблице 15:

Таблица 15 – Основные показатели экономической эффективности

|  |  |
| --- | --- |
| Показатель | Значение |
| Ожидаемый годовой экономический эффект, тг |  |
| Коэффициент экономической эффективности капитальных вложений | 0,58 |
| Срок окупаемости капитальных вложений, месяцев | 21 |

# Вывод

В этой дипломной работе были проведены и вычислены затраты на разработку приложения. Разрабатываемое ПО обеспечит экономию рабочего времени заведующего учебной части за счёт уменьшения количества бумажной работы и автоматизации процесса распределения нагрузки на преподавателей, а также снижение количества ошибок, так как вся информация будет в электронном виде и ее можно будет легко выгрузить ответственному лицу.

Большую часть затрат на разработку приложения составят затраты на покупку девайсов и приобретение ПО (83% от общих расходов).

Согласно произведённым расчётам, цена реализации разрабатываемого программного продукта с учётом НДС составит ‬ тенге, себестоимость ПО (смета затрат) – 843102 тенге. Соответственно, ожидаемая прибыль от реализации ПО составит тенге.

Ожидаемая условно годовая экономия составит тенге. Приложение окупится в первые 21 месяцев его использования.

# 5 Безопасность жизнедеятельности

# Анализ условий труда в офисе учебной части РФМШ, расчет производственного освещения

Моя дипломная работа представляет из себя приложение для заведующего учебной частью человека, в котором можно будет автоматически распределяя нагрузку на преподавателей создать расписание, в котором каждый может выгрузить ту информацию, которая ему необходима.

Анализ условий труда будет предоставляться по школе НАО «РФМШ», в которой я работаю и по сей день. Здание компании расположено по адресу г. Алматы, улица Бухар-Жырау, дом 36.

Кабинет заведующего учебной частью имеет площадь 20 м2, площадь окна занимает 2,0 м2. В качестве источников искусственного освещения используются 1 светильник с 1 светодиодной лампой типа Standart, мощностью 40 Вт, напряжением 230 В и световым потоком 370 лм, также присутствует вентиляция в виде кондиционера. В этом кабинете имеется 2 рабочих места, а также предоставлено следующее оборудование для работы: ПК (HP Bundle) – 2 единицы, МФУ (Canon mf421dw) – 2 единицы. В постоянном штате на рабочем месте присутствует два человека.

Вышеописанное оборудование не превышает предельно-допустимые нормы по шуму, и внешний шум также не является недопустимым, так как окна рабочего кабинета выходят на тихую часть нашего кампуса, где обучается в основном старшая параллель (10-11 класс). Беря во внимание вышеописанное, освещенность в комнате не соответствует допустимой норме, также взяв во внимание наличие маленькой площади окна и учитывая географические расположение корпуса, в котором оно находится в следствие из этого солнце редко светит в эту комнату. Вентиляция проложена обеспечивая 10 м3/ч на человека.

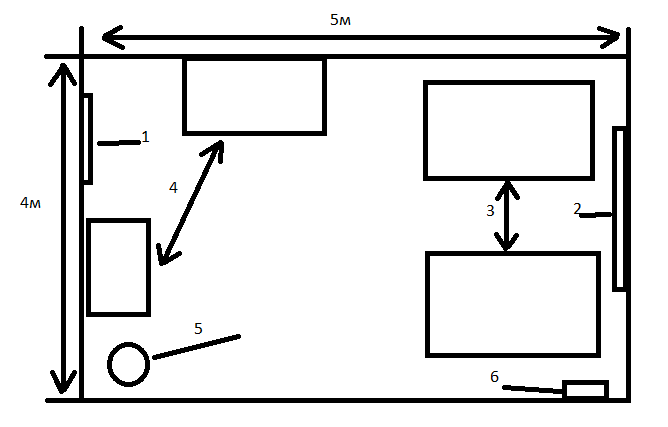


Рисунок 27 – План рабочего помещения

Условные обозначения: 1 – дверной проем; 2 – оконный проем; 3 – рабочее место; 4 – шкаф; 5 – огнетушитель;

Но в кабинете, в котором составляется расписание занятий для всей школы освещение не соответствует нормам труда, в связи с этим необходимо произвести проверку освещения и постараться немного улучшить его до стандартов.

Комната имеет размеры: 5x4x3. Высота рабочей поверхности над уровнем пола 1,5 м.

# 5.2 Расчет естественного освещения

При боковом освещении определяют площадь световых проемов S, обеспечивающую нормированные значения КЕО, по формуле:

, (5.1)

Из формулы (1.1) определим площадь светового проёма:

, (5.2)

где Sn – площадь помещения, м2;

en – нормированное значение КЕО;

– коэффициент запаса учитывает запыленность помещения снижение светового потока ламп в процессе эксплуатации;

kЗД – коэффициент, учитывающий затенение окон противостоящими зданиями.

– общий коэффициент светопропускания;

– световая характеристика окон;

– коэффициент, учитывающий повышение КЕО при боковом освещении благодаря свету, отраженному от поверхностей помещения и подстилающего слоя, прилегающего к зданию.

(5.3)

Найдем значение по формуле (5.4):

- общий коэффициент светопропускания равный:

tо =  t1 \*t2 \*t3 \*t4, (5.4)

где   t1 - коэффициент светопропускания материала;

t2 - коэффициент, учитывающий потери света в переплетах светопроема;

t3 - коэффициент, учитывающий потери света в несущих конструкциях, при боковом освещении равен 1

t4 - коэффициент, учитывающий потери света в солнцезащитных устройствах;

;

- световая характеристика окон;

Отношение длины комнаты к глубине наиболее удалённой точки от окна равно . Отношение ширины помещения к высоте от уровня рабочей поверхности до верха окна равно . Отсюда .

 - коэффициент, учитывающий повышение КЕО при боковом освещении благодаря свету, отражённому от поверхностей помещения и подстилающего слоя, прилегающего к зданию;

kЗД – коэффициент, учитывающий затенение окон противостоящими зданиями.

Поскольку затеняющих зданий поблизости нет, то kЗД=1.

Вычислим общую площадь окон:

.

По расчетам получили, что, согласно санитарным нормам, общая площадь окон должна быть 5,43 м2. Существующего бокового освещения от оконного проема площадью каждого S = 2м2 недостаточно.

# 5.3 Проверка искусственного освещения

Рассчитаем требуемое рабочее освещение для помещения со следующими габаритными размерами: длина А = 5 м, ширина В = 4 м, высота Н = 3 м. Высота рабочей поверхности над уровнем пола 1,5 м.

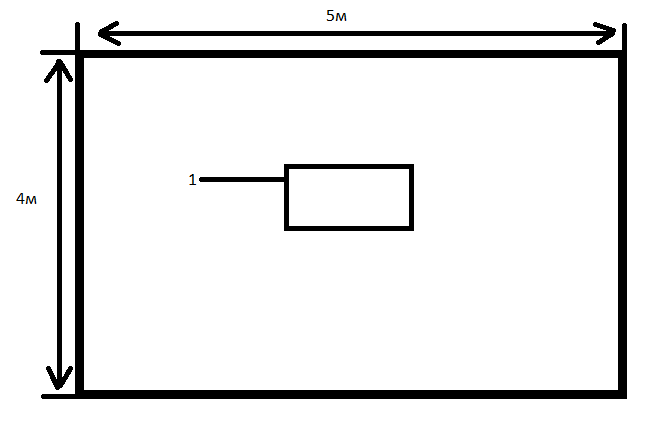


Рисунок 28 – План светильников

Проверяем соответствие существующего искусственного освещения нормативному, расчетом освещенности точечным методом:

 (5.5)

где μ –коэффициент, учитывающий освещенность от удаленных светильников и отраженный световой поток от стен, потолка и расчетной поверхности;

Iα- сила света в направлении рассматриваемой точки;

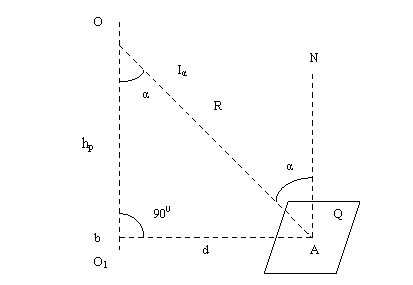


Рисунок 29 – Схема к расчету освещенности точечным методом

Горизонтальная освещенность в точке *А* от одного светильника определяется следующей формулой:

(5.6)

(5.7)

hр= Н -(hраб. +hсв) (5.8)

Н = 2,8 м;

hраб = 1,5 м;

hсв. = 0,5 м.

h р = 3 - (1,5 + 0,5) = 1 м.

d = 0 м;

Ia=175 кд при 0о по таблице (3.10) [7]

Рассчитаем освещенность горизонтальной плоскости в контрольной точке:

μ = 1,3 - 1,6;

К3 = 1.5 – по таблице (3.11) [7];

Fл = 370 лм.

Emin = 200, соответственно ЕАГ меньше Emin.

Существующее на данный момент искусственное освещение в этой комнате не соответствует нормативному, исходя из этого нужно провести реконструкцию освещения.

Теперь надо провести реконструкцию искусственного освещения.

# 5.4 Расчет освещения методом коэффициента использования светового потока

Производим реконструкцию искусственного освещения методом коэффициента использования светового потока

 (5.9)

*N -*число ИС;

*К -*коэффициент запаса;

z - коэффициент минимальной освещенности (отношение средней и минимальной ос­вещенности).

В расчетах коэффициент z принимается равным: 1,15 - для светильников, располагаемых по вершинам прямоугольных полей

Следовательно, для обеспечения достаточной освещенности необходимо модифицировать систему путем установки более мощных ламп. Заменим наши лампы STANDARD 40W 230V со световым потоком 370 лм на лампы HPL-N 50W со световым потоком 3200 лм.

Сделаем расчет для новых ламп:

α=700

Ia=53,73 кд при 700 по таблице (3.10)

μ = 1,3 - 1,6;

К3 = 1.5 – по таблице (3.11);

Fл = 3200 лм.

Предложенная система освещения обеспечивает нормированную освещенность, так как ЕАГ > Eн. (213,97 > 200) , что составляет 114% от нормы.

# 5.5 Эргономические требования к рабочему месту за ПК.

Развитие информационных технологий предало значение компьютерной эргономике.

В настоящее время имеется несколько понятий эргономики:

1. Эргономика — это наука о приспособлении должностных обязанностей, рабочих мест, предметов и объектов труда, а также компьютерных программ для наиболее безопасного и эффективного труда работника, исходя из физических и психических особенностей человеческого организма.
2. Более широкое определение эргономики, принятое в 2010 году Международной ассоциацией эргономики эргономика - это научная дисциплина, изучающая взаимодействие человека и других элементов системы, а также сфера деятельности по применению теории, принципов, данных и методов этой науки для обеспечения благополучия человека и оптимизации общей производительности системы.

Можно сказать, что человеческий фактор определяет успех любой компании. От того, как можно комфортнее сотрудник чувствует себя на рабочем месте, зависит эффективность и производительность его работы, а в следствии и всей компании. А удобство во многом определяется тем, в каких условиях проходит трудовой процесс.

Основной принцип организации рабочего места – минимизация нагрузок, удобство и комфортность. Характеристика эргономики рабочего места определяется психологическими, физиологическими и антропометрическими требованиями. В соответствии с этим учитывается рабочая поза; возможность охвата движениями и взглядом всего пространства и расположенных на нем предметов; пространство, на котором размещается сам работник; возможность работы с техникой, ведения записей, размещения необходимых материалов.

В соответствии с требованиями эргономики, компьютер должен располагаться в помещении так, чтобы свет из окна падал слева, искусственное освещение должно быть равномерным, конструкция рабочего стола 727 мм, стул подъемно-поворотный, верхняя часть экрана должна располагаться на уровне глаз, расстояние от монитора до глаз 60-80 см, поверхность клавиш вогнутая, длина пространства для ног 450 мм, высота пространства для ног 600мм, ширина пространства для ног 500 мм.

Характеристика рабочего места с ПК любого специалиста должна соответствовать определенным правилам и учитывать: антропометрическую совместимость – соответствие размеров тела и его положения при работе; сенсомоторную совместимость – скорость моторных операций; энергетическую совместимость – усилия, прилагаемые для выполнения действий; психофизиологическую совместимость – реакцию на внешние эстетические параметры.

Кроме формы и габаритов мебели, расположения человека относительно техники и размещения клавиатуры, монитора и других частей рабочего процесса, эргономика рабочего места за компьютером должна учитывать размеры и форму периферии.

Компьютерная эргономика исповедует три взаимосвязанных принципа:

1. *Альфа принцип* - унификация или следование общепринятому образцу, позволяет освоить стандартные действия, экономить время обучения пользователя.
2. *Бета принцип* — это настраиваемость под нужды и привычки конкретного пользователя (регулирование чувствительности и интервал между нажатиями для двойного щелчка, возможность запрограммировать клавиши на клавиатуре).
3. *Гамма принцип* - предполагает оперативный учет контекста в операционных системах и прикладных программах, имеется контекстное меню и справки, одни и те же средства управления в зависимости от условий выполняемых действий.

Во взаимодействии между человеком и машиной и создании новых устройств обязательно в настоящее время участвуют психологи, которые знают, что люди в восприятии переработки информации делятся на: визуалов, которым более удобны разнообразные знаки, пиктограммы; аудиалов – текстовые надписи; кинестетиков, которые придают значение телесным ощущениям, они особо чувствительны к месту положения клавиш и экранных кнопок, привередливы к тактильным свойствам клавиатуры и мыши.

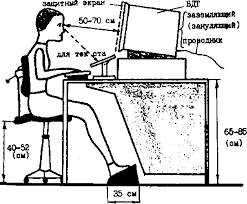


Рисунок 30 – Правильная посадка за рабочее место ПК-пользователя.

Следует отметить, что при не выполнении эргономических требований развиваются множество болезней симптомами которых являются боль и дискомфорт в пояснице, которые возникают из-за неправильного положения спины, сутулости, неправильного положения ног; из-за неправильного положения рук на клавиатуре или мыши могут серьезно пострадать кисти, запястья и предплечья (самым распространенным заболеванием является кистевой туннельный синдром); приходится наклонять спину, шею при работе с монитором и документами, что приводит к повышенным нагрузкам и напряжениям мышц, и вызывает боль и дискомфорт мышц спины, шеи и плечевой части корпуса.

Таким образом, если будет учтена эргономика при работе за компьютером, правильно обустроена рабочее место и пространство рабочего места, исчезнут детали, доставлявшие дискомфорт и мешавшие работе любого специалиста. Такие изменения позволят как увеличить эффективность, так и сэкономят время, которое тратилось на самостоятельное преодоление пусть даже незаметных сразу, но важных «мелочей».

# Вывод

В результате анализа условий работы в школе НАО «РФМШ» города Алматы, где я проходил преддипломную практику и параллельно писав дипломную работу, были выявлены проблемы с освещением, а также с эргономикой помещения.

Был проведен расчёт производственного освещения, которое необходимо для комфортной работы в кабинете заведующего учебной частью. Ввиду недостаточной освещенности комнаты, которую смогли выявить с помощью расчетов, произвели реконструкцию производственного освещения.

# Заключение

Таким образом, в результате работы были решены поставленные задачи и получены следующие результаты:

* исследованы деятельность и организационная структура НАО «РФМШ», на основании чего был сделан вывод о необходимости повышения эффективности работы школы в целом и заместителя директора по УВР в части автоматизации информационного процесса составления расписания;
* проведен анализ существующей организации информационных процессов составления расписания в школе и выявлены недостатки, устранение которых возможно за счет внедрения автоматизированной системы составления расписания;
* составлена спецификация и проведено обоснование функциональных и нефункциональных требований к разрабатываемой автоматизированной системе составления расписания, учитывающих нормативно-правовую документацию и специфические особенности учебной деятельности школы;
* спроектированы база данных и автоматизированная система составления расписания в школе, учитывающие ранжирование трудности предметов и методические дни учителей;
* разработано программное обеспечение, внедрение которого позволяет обеспечить повышение эффективности и снижение трудозатрат персонала РФМШ посредством автоматизации информационных процессов составления расписания в школе;
* оценена эффективность предложенного проекта, в результате чего был сделан вывод об экономической целесообразности разработки и внедрения программного продукта.

Разработанная автоматизированная система составления расписания в школе может быть адаптирована и тиражирована практически для любых учебных учреждений, коммерческих и некоммерческих организаций, одним из направлений которых является составление различных расписаний событий и мероприятий с целью снижения издержек и повышения эффективности работы. При этом необходимо отметить, что в рамках дальнейшей доработки предложенного проекта рекомендуется учесть функциональную возможность закрепления нескольких уроков за одним учителем.

# Список использованной литературы

1. Г. Боканова Методические указания по выполнению экономической части дипломных работ Алматы, АУЭС, 2019 – 33с.
2. Технико-экономическое обоснование дипломных проектов Брест, БГТу, 2014 – 15с.
3. Методические указания к выполнению расчетно-графической работы для студентов специальности 080801 «Прикладная информатика (по областям)» Уфа, 2012 – 12с.
4. Методические указания по выполнению экономической части дипломных работ Москва, Московский университет им. С.Ю. Витте, 2016 – 21с.
5. Симионов Ю.Ф., Боромотов В.В. Информационный менеджмент. — Ростов н.Д: Феникс, 2013, 250с.
6. Дубовцев В.А. Безопасность жизнедеятельности. / Учеб. пособие для дипломников. - Киров: изд. КирПИ, 2013.
7. Ротер М. Учитесь видеть бизнес–процессы: Построение карт по-токов создания ценности / М. Ротер. – М.: Альпина Паблишер, 2015. – 136 c.
8. Тельнов Ю. Ф. Реинжиниринг бизнес–процессов. Компонентная методология [Текст] / Ю. Ф. Тельнов. Изд. 2–е, перераб. и доп. М.: Финансы и статистика, 2014. – 320 с.
9. Шерегов Н.А. Теоретические и прикладные аспекты моделирова-ния деятельности компании// электронный научный журнал// Управление эко-номическими системами, 2013 год – С. 54–58.
10. Гигиенические требования к условиям обучения в образователь-ных учреждениях СанПиН 2.4.2.2821-10 // [Электронный ресурс] Режим до-ступа: http://base.garant.ru/12183577/ (Дата обращения: 23.01.2021)
11. Елиферов В.Г. Бизнес–процессы: Регламентация и управление: Учебник / В.Г. Елиферов. – М.: НИЦ ИНФРА–М, 2013. – 319 c.
12. Репин В.В. Бизнес–процессы. Моделирование, внедрение, управление / В.В. Репин. – М.: Манн, Иванов и Фербер, 2013. – 512 c.
13. Долганова О.И. Моделирование бизнес–процессов: Учебник и практикум для академического бакалавриата / О.И. Долганова, Е.В. Виногра-дова, А.М. Лобанова. – Люберцы: Юрайт, 2016. – 289 c.
14. Дейт, К. Введение в системы баз данных; К.: Диалектика; Издание 6–е, 2012. – 360 c.
15. Туманов В.Е. Основы проектирования реляционных баз данных; Бином, 2012. – 420 c.
16. Mistry R. Introducing Microsoft SQL Server 2014. – 541p.
17. Chowdhury K. Mastering Visual Studio 2017 –2017. – 466 p.
18. Биллиг, В.А. Основы программирования на С#: Учебное пособие / В.А. Биллиг. – М.: Бином, 2012. – 483 c.
19. ГОСТ 34.003-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения //[Электронный ресурс] Режим доступа: http://base.garant.ru/187632/ (Дата обращения: 23.01.2021)
20. Громов А.И. Управление бизнес–процессами: современные мето-ды. монография / А.И. Громов, А. Фляйшман, В. Шмидт. – Люберцы: Юрайт, 2016. – 367 c.
21. Чукарин А.В. Бизнес-процессы и информационные технологии в управлении современной инфокоммуникационной компанией / А.В. Чукарин. - М.: Альпина Паблишер, 2016. - 512 c.
22. Репин В.В. Процессный подход к управлению. Моделирование бизнес–процессов / В.В. Репин. – М.: Манн, Иванов и Фербер, 2013. – 544 c.

**Приложение А**

**Техническое задание**

Техническое задание на разработку автоматизированной системы составления расписания для филиала города Алматы НАО «РФМШ»»

1 Общие требования:

Наименование разрабатываемого приложения:

Расписание НАО «РФМШ».

Цели разработки:

* повышение эффективности работы;
* снижение трудозатрат;
* оптимизация составления расписания;

2 Технические требования:

Требования к структуре и функционированию системы:

* приложение должно работать без зависаний;
* программа должна работать как на старых машинах, так и на новых
* обеспечение защиты данных пользователей.

Требования к функциям, выполняемым системой:

* выгрузка расписания по всей школе;
* добавление и изменение преподавателей;
* добавление и изменение классов;
* добавление и изменение нагрузки на класс

Требования к интерфейсу системы:

* дружелюбный интерфейс;
* удобство, интуитивная понятность;
* админ панель для редактирования данных.

# Приложение Б

(листинг программы)

<div class="row" \*ngIf="viewMode">

    <div class="col-md-12">

        <div class="form-group text-center">

            <label style="font-size: 1.3rem;"><b>{{lecture.name}}</b></label>

            <!-- <input class="form-control" [(ngModel)]="lecture.name" [readonly]="viewMode"> -->

        </div>

    </div>

</div>

<div class="row">

    <div class="col-md-12" \*ngIf="viewMode && lecture.subjectId != null">

        <!-- <div class="form-group"> -->

            <label style="font-size: 1rem;"><b>Навзание предмета:</b> {{lecture.subjectName}}</label>

            <!-- <input class="form-control" [(ngModel)]="lecture.subjectName"

             [readonly]="viewMode"> -->

        <!-- </div> -->

    </div>

    <div class="col-md-12" \*ngIf="!viewMode">

        <div class="form-group">

            <label style="font-size: 1rem;">Прикрепить к предмету. Название предмета: </label>

            <div class="input-group">

                <input class="form-control" [(ngModel)]="lecture.subjectName"

                (click)="openSelectSubjectModal()" readonly>

             <span class="input-group-btn">

                <button (click)="openSelectSubjectModal()" class="btn btn-outline-primary"

                    type="submit"> <i class="fa fa-search"></i> Выбрать</button>

            </span>

            <span class="input-group-btn">

                <button (click)="deleteSubject()" class="btn btn-outline-danger"

                    type="submit"> <i class="fa fa-trash"></i> Удалить</button>

            </span>

        </div>

        </div>

    </div>

</div>

<div class="row" \*ngIf="!viewMode">

    <div class="col-md-12">

        <div class="form-group">

            <label style="font-size: 1rem;">Название лекции: </label>

            <input class="form-control" [(ngModel)]="lecture.name" [readonly]="viewMode">

   </div>

    </div>

</div>

*Продолжение приложения Б*

<div class="row pb-2" \*ngIf="!viewMode">

    <div class="col-md-12">

        <div class="form-group">

            <label style="font-size: 1rem;">Контент лекции: </label>

            <angular-editor [placeholder]="'Enter text here...'" [(ngModel)]="lecture.content" [config]="editorConfig"></angular-editor>

        </div>

        <!-- <angular-editor [placeholder]="'Enter text here...'" [(ngModel)]="lecture" [config]="editorConfig"></angular-editor> -->

    </div>

</div>

<div class="row pb-2" \*ngIf="viewMode">

    <div class="col-md-12">

        <label style="font-size: 1rem;"><b>Контент лекции:</b> </label>

        <div class="col-md-12 border-top border-bottom">

            <div style="font-size: 1rem;" [innerHtml]="lecture.content"></div>

            <!-- <p>{{lecture}}</p> -->

        </div>

    </div>

</div>

<div class="row" \*ngIf="!viewMode">

    <div class="col-md-12">

        <div class="form-group">

            <label style="font-size: 1rem;">Загрузить файлы лекции: </label>

            <app-file-upload  (onUploadFinished)="uploadFinished($event)" (fileName)="getFileName($event)"></app-file-upload>

        </div>

    </div>

</div>

<div class="row" \*ngIf="lectureFiles.length != 0">

    <div class="col-md-12">

        <label style="font-size: 1rem;" [class.boldStyle]="viewMode">Прикрепленные файлы к лекции:</label>

    </div>

</div>

<div class="container">

    <ul>

        <li \*ngFor="let file of lectureFiles">

            <a>

                {{file.fileName}}

                <app-download [fileName]="file.fileName"></app-download>

                <button \*ngIf="!viewMode" class="btn btn-outline-danger ml-2" (click)="delete(file.fileName)"><i class="fa fa-trash"></i> Удалить</button>

            </a>

        </li>

*Продолжение приложения Б*

    </ul>

</div>

<div class="row">

    <div class="col-md-12" \*ngIf="viewMode && lecture.testId != null">

        <!-- <div class="form-group"> -->

            <label style="font-size: 1rem;"><b>Навзание теста:</b> {{lecture.testName}}</label><br>

            <!-- <input class="form-control" [(ngModel)]="lecture.testName"

             [readonly]="viewMode"> -->

        <!-- </div> -->

        <button \*ngIf="showLink" class="btn btn-md btn-outline-primary" (click)="RedirectToTest()"><i class="fa fa-external-link"></i> Перейти к тестированию</button>

        <button \*ngIf="!showLink" class="btn btn-md btn-outline-primary" (click)="viewTestModal.show(lecture.testId)"><i class="fa fa-eye"></i> Просмотр теста</button>

    </div>

    <div class="col-md-12" \*ngIf="!viewMode">

        <div class="form-group">

            <label style="font-size: 1rem;">Прикрепить тест. Название теста: </label>

            <div class="input-group">

                <input class="form-control" [(ngModel)]="lecture.testName"

                (click)="openSelectTestModal()" readonly>

             <span class="input-group-btn">

                <button (click)="openSelectTestModal()" class="btn btn-outline-primary"

                    type="submit"> <i class="fa fa-search"></i> Выбрать</button>

            </span>

            <span class="input-group-btn">

                <button (click)="deleteTest()" class="btn btn-outline-danger"

                    type="submit"> <i class="fa fa-trash"></i> Удалить</button>

            </span>

        </div>

        </div>

    </div>

</div>

<div class="row">

    <div class="col-md-12" \*ngIf="viewMode && lecture.practiceId != null">

        <!-- <div class="form-group"> -->

            <label style="font-size: 1rem;"><b>Навзание практического задания:</b> {{lecture.practiceName}}</label> <br>

            <!-- <input class="form-control" [(ngModel)]="lecture.practiceName"

             [readonly]="viewMode"> -->

        <!-- </div> -->

*Продолжение приложения Б*

   <button \*ngIf="showLink" class="btn btn-md btn-outline-primary" (click)="RedirectToPractice()"><i class="fa fa-external-link"></i> Перейти к практическому заданию</button>

        <button \*ngIf="!showLink" class="btn btn-md btn-outline-primary" (click)="viewPracticeModal.show(lecture.practiceId)"><i class="fa fa-eye"></i> Просмотр практического задания</button>

    </div>

    <div class="col-md-12" \*ngIf="!viewMode">

        <div class="form-group">

            <label style="font-size: 1rem;">Прикрепить практическое задание. Название практического задания: </label>

            <div class="input-group">

                <input class="form-control" [(ngModel)]="lecture.practiceName"

                (click)="openSelectPracticeModal()" readonly>

             <span class="input-group-btn">

                <button (click)="openSelectPracticeModal()" class="btn btn-outline-primary"

                    type="submit"> <i class="fa fa-search"></i> Выбрать</button>

            </span>

            <span class="input-group-btn">

                <button (click)="deletePractice()" class="btn btn-outline-danger"

                    type="submit"> <i class="fa fa-trash"></i> Удалить</button>

            </span>

        </div>

        </div>

    </div>

</div>

<div class="row pt-4">

    <div class="col-md-12">

        <div class="d-flex justify-content-center">

            <button class="btn btn-md btn-outline-primary mr-5"  (click)="goBack()"><i class="fa fa-arrow-left"></i> Назад</button>

            <button  \*ngIf="!viewMode" class="btn btn-outline-primary px-3 py-2" (click)="create()"><i class="fa fa-save"></i> Сохранить</button>

        </div>

    </div>

</div>

<app-test-lookup-table-modal #testLookupTableModal (modalSave)="selectTest()"></app-test-lookup-table-modal>

<app-subject-lookup-table-modal #subjectLookupTableModal (modalSave)="selectSubject()"></app-subject-lookup-table-modal>

<app-practice-lookup-table-modal #practiceLookupTableModal (modalSave)="selectPractice()"></app-practice-lookup-table-modal>

<app-view-practice-modal #viewPracticeModal></app-view-practice-modal>

<app-view-test-modal #viewTestModal></app-view-test-modal>