|  |  |
| --- | --- |
| Утверждаю  Руководитель (должность, наименование предприятия- заказчика)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ФИО “\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г. | Утверждаю Руководитель (должность, наименование предприятия- разработчика)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ФИО “\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г. |

\_\_\_\_\_\_\_Система по автоматизации процесса составления\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_расписания на факультете\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Наименование продукта

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Система составления расписания \_\_\_\_\_\_\_\_\_\_\_\_  
Сокращенное наименование продукта

Техническое задание

На 10 листах

Действует с 27.03.2019 г.

|  |  |
| --- | --- |
| Согласовано  Руководитель  Полещук Х.А  27.03.2019 г. | Составили Студентки 3 курса 3.2 группы  Дьяконова М.А  Пастухова О.В  25.03.2019 г. |

Воронеж 2019

Содержание

[**Общие сведения** 3](#_Toc10293740)

[1. Общие положения 4](#_Toc10293741)

[1.1 Наименование заказчика 4](#_Toc10293742)

[1.2. Наименование исполнителя 4](#_Toc10293743)

[1.3 Наименование системы 4](#_Toc10293744)

[1.4 Плановые сроки начала и окончания работы 4](#_Toc10293745)

[2. Назначение и цели создания системы 5](#_Toc10293746)

[2.1. Назначение системы 5](#_Toc10293747)

[2.2 Цель создания системы 5](#_Toc10293748)

[3. Требования к системе 6](#_Toc10293749)

[3.1 Общие требования 6](#_Toc10293750)

[3.2 Функциональные требования 6](#_Toc10293751)

[3.3 Требования к безопасности 7](#_Toc10293752)

[3.4 Требования к программному обеспечению 7](#_Toc10293753)

[3.5 Требования к качеству 8](#_Toc10293754)

[3.6 Дизайн интерфейса системы 8](#_Toc10293755)

[4. Состав и содержание работ по созданию системы 10](#_Toc10293756)

[6. Риски 10](#_Toc10293757)

[7. Порядок контроля и приемки системы 11](#_Toc10293758)

[8.Требования к документированию 11](#_Toc10293759)

[**Приложение** 12](#_Toc10293760)

# **Общие сведения**

Настоящие документ является Техническим Заданием к проекту «Система по автоматизации процесса составления расписания». В него входят общие сведения о проекте, цели разработки, требования, информация об интерфейсе. По ходу разработки документ может изменяться и расширяться.

Для конкретного понимания темы работы составлен небольшой справочник узкоспециализированных слов, которые используются в Техническом Задании. Ниже представлены понятия и определения к ним.

1. Расписание - вид календаря (т.е. упорядоченность по времени), для которого указана информация о предстоящих (планируемых) событиях. Обычно представляется в виде таблицы.
2. Аудитория - помещение для проведения лекций. Не оборудовано компьютерной и специализированной техникой.
3. Лаборатория - помещение для проведения практических работ. Оборудовано специализированной техникой (компьютерами).
4. Преподаватель - человек, педагог, который занимается преподаванием какого-либо предмета. Работник вуза.
5. Лабораторные занятия (практические) - занятия в отдельных группах. Проведение учащимися по заданию преподавателя опытов с использованием спецоборудования. Проводятся в лабораториях.
6. Группа студентов - учётная единица. Характеризуется названием (номером) и относительно постоянным составом.
7. Учебная пара - аудиторное занятие, происходит в виде двух совмещённых академических часов.

# Общие положения

## 1.1 Наименование заказчика

Ассистент Тарасов Вячеслав Сергеевич. Кафедра программирования и информационных технологий.

## 1.2. Наименование исполнителя

Студент Дьяконова Мария Александровна. Кафедра информационных технологий управления.

Студент Пастухова Ольга Васильевна. Кафедра информационных технологий управления.

## 1.3 Наименование системы

Полное наименование системы: Система по автоматизации процесса составления расписания на факультете.

Краткое наименование: Система составления расписания.

## 1.4 Плановые сроки начала и окончания работы

Плановый срок начала работ – 1.03.2019 г.

Плановый срок окончания работ – 20.06.2019 г.

# 2. Назначение и цели создания системы

## 2.1. Назначение системы

Основное назначение данной системы - повысить эффективность составленного расписания, которое напрямую влияет на качество обучения, путем автоматизации основных процессов, таких как создание списка необходимых предметов для каждой группы каждого курса, создание списка предметов для каждого преподавателя. На основе этих списков вручную формируется расписание.

## 2.2 Цель создания системы

Цель работы – реализовать приложение для составления расписания занятий в учебных заведениях, которое позволит учесть все требования и пожелания, предъявляемые к нему, сократить время, потраченное на его составление, сделать процесс создания расписания более удобным.

В результате формирования расписания должны быть улучшены значения следующих показателей:

* Время, потраченное на получение и подготовку исходных данных для составления расписания, таких как получение списка предметов для каждой группы и списка предметов для каждого преподавателя, а также списка доступных аудиторий для факультета.
* Время, потраченное на составление расписания в целом.
* Распределение пар между группами, минимизация окон между парами.

# 3. Требования к системе

## 3.1 Общие требования

* Использование информации о предметах и преподавателях из учебных планов специальностей.
* Использование информации о доступных аудиториях для факультета.
* Расширяемость системы (возможность её доработки в случае повышения требований к автоматизированной системе).
* Наличие логичного и понятного пользовательского интерфейса, обеспечивающего легкое восприятие и обработку информации, а также минимум операций пользователя.

## 3.2 Функциональные требования

Данная система предназначена для составления расписания одним администратором. В других действующих лицах нет необходимости, так как они будут излишними. После процесса составления расписания, администратор экспортирует расписание на компьютер для того, что бы его уже могли просмотреть другие пользователи.

Функции, которые должна выполнять система:

* Хранение всей необходимой информации о предметах, преподавателях, аудиториях и группах студентов в базе данных.
* Получение всех необходимых таблиц из БД при запуске приложения.
* Возможность внесения изменений в существующую базу данных, а именно удаление из базы данных, изменение и добавление данных в таблицах.
* Проверка на корректность ввода при добавлении новых данных, т.е. при добавлении преподавателя или названия предмета может использоваться только русский и английский алфавит, цифры и иные символы не должны быть добавлены.
* Возможность вывода свободных аудиторий для занятия (можно сделать вывод всех аудиторий,а требование вывода свободных аудиторий может быть выполнено при необходимости дальнейшего расширения системы).
* Возможность вывода списков всех групп на каждом курсе, списка предметов для каждой группы.
* Возможность вывода всех преподавателей и предметов, соответствующих каждому преподавателю.
* Возможность вывода всех аудиторий, доступных факультету.
* Экспорт расписания из приложения в MicrosoftExcel.

## 3.3 Требования к безопасности

При разработке программного кода разработчик должен принять методы безопасного программирования, включающие:

* Проведение ручной проверки кода на предмет незадекларируемых возможностей.
* Использование при разработке доверенной аппаратной платформы с функциями защиты от незадекларированных возможностей на системном и прикладном уровне.
* Программное обеспечение должно предусматривать базовую защиту от основных видов атак: SQL-инъекций.

## 3.4 Требования к программному обеспечению

В качестве инструмента для визуального проектирования БД была выбрана MySQL Workbench, так как она более удобна в использовании, чем, например, DBDesigner и другие. Этот продукт позволяет создавать новую БД, вносить изменения в уже существующую, выполнять запросы в реальном времени и наглядном виде. Также все связи между таблицами показываются на ER-диаграмме, что тоже очень удобно.

В качестве платформы для веб-разработки используется ASP.NET. Ее основными преимуществами является расширяемый набор элементов управления и библиотек классов, что позволяет быстрее разрабатывать приложение. ASP.Net предоставляет возможность разделения визуальной части и бизнес-логики по разным файлам.

## 3.5 Требования к качеству

* Система должна быть функционально исправна, реализация основных функций(добавление/удаление/изменение предметов, преподавателей, групп в соответствующих таблицах, получение списка всех аудиторий, возможность на основе данных таблиц составления расписания в ручную).
* Эффективность использования во времени.
* Запуск системы на ОС Windows 8 и выше не более чем за 3 секунды.
* Удобство использования, понятность, наличие сообщений-подсказок.
* Выход из тестирования должен быть не меньше чем с 90% пройденных тест-кейсов, не пройдённые тест-кейсы не должны влиять на работоспособность системы.

## 3.6 Дизайн интерфейса системы

На главном экране разрабатываемого приложения могут быть следующие кнопки, которые вызывают соответствующие экраны:

1. Показать расписание (Рисунок 1)
2. Показать список преподавателей (Рисунок 2)
3. Показать список учебных групп(Рисунок 3)
4. Показать список предметов(Рисунок 4)

В процессе разработки дизайн может меняться.



Рисунок 1

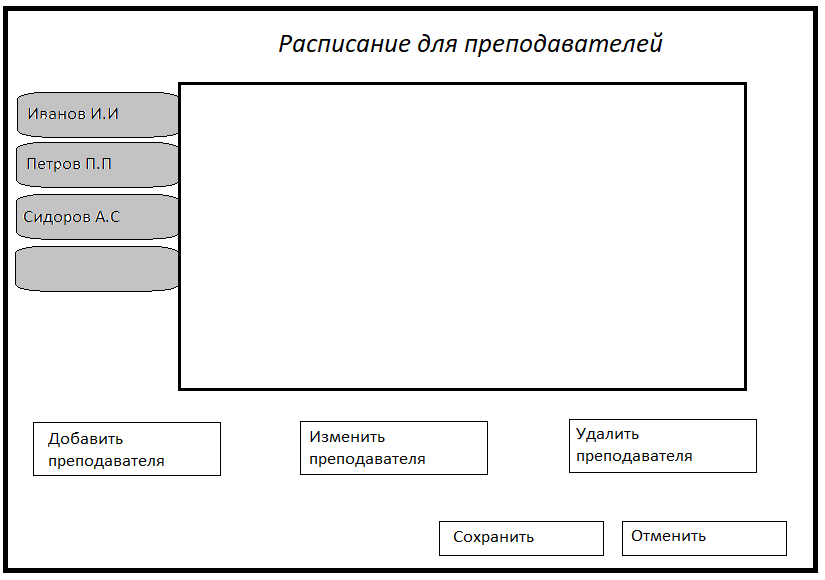


Рисунок 2

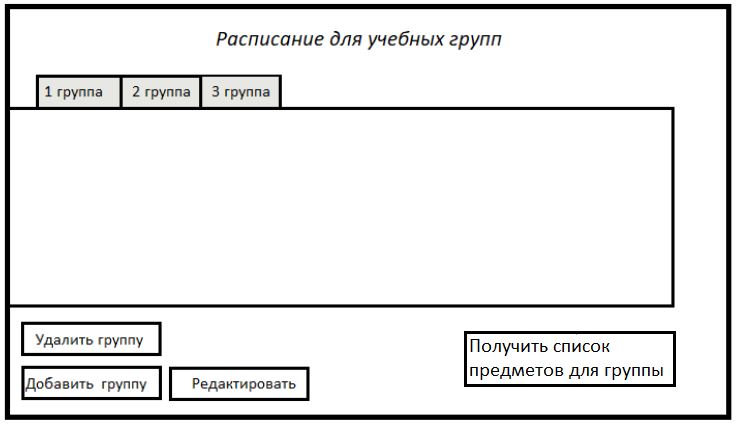


Рисунок 3

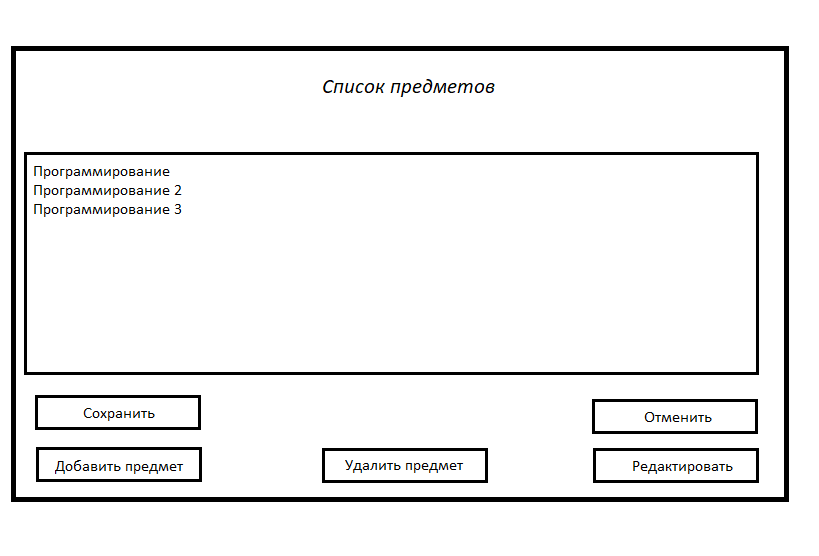


Рисунок 4

# 4. Состав и содержание работ по созданию системы

1. Допроектное обследование, сбор необходимой информации, для определения целей и задач системы, которые в дальнейшем надо решить.
2. Анализ предметной области и введение организационных требований к решению задач и целей.
3. Разработка модели программы: построение модульной схемы системы, определение связей между сущностями, построение концептуальной модели БД.
4. Разработка рабочего проекта, состоящего из:

* написания программы;
* отладка программы;
* корректировка программы;

6. Проведение тестирования и доработка информационного программного обеспечения по замечаниям и предложениям.

7. Сдача системы в эксплуатацию.

# 6. Риски

* Нечитаемый код, что может помешать дальнейшей расширяемости системы.
* Неэффективный выбор методов тестирования, что влечет за собой невыполнение основных требований к приложению.
* Неудобный пользовательский интерфейс.

# 7. Порядок контроля и приемки системы

* Предварительные испытания.

Участники испытаний: Разработчики(Пастухова О.В, Дьяконова М.А)

Срок проведения с 20.04.2019 по 1.05.2019 г.  
 Проведение предварительных испытаний. Устранение выявленных неполадок. Проверка устранения выявленных неполадок. Принятие решения о возможности передачи программного обеспечения в опытную эксплуатацию.

* Опытная эксплуатация

Участники испытаний: Разработчики(Пастухова О.В, Дьяконова М.А)  
Срок проведения с 1.05.2019 по 15.05.2019 г.

Проведение опытной эксплуатации. Устранение выявленных неполадок. Проверка устранения выявленных неполадок. Принятие решения о готовности программного обеспечения к приемочным испытаниям.

* Приемочные испытания

Участники испытаний: Разработчики(Пастухова О.В, Дьяконова М.А) Заказчик(Тарасов В.С.)

Срок проведения с 1.06.2019 по 20.06.2019 г.

Проведение приемочных испытаний. Принятие решения о возможности передачи программного обеспечения в промышленную эксплуатацию. Оформление Акта завершения работ.

# 8.Требования к документированию

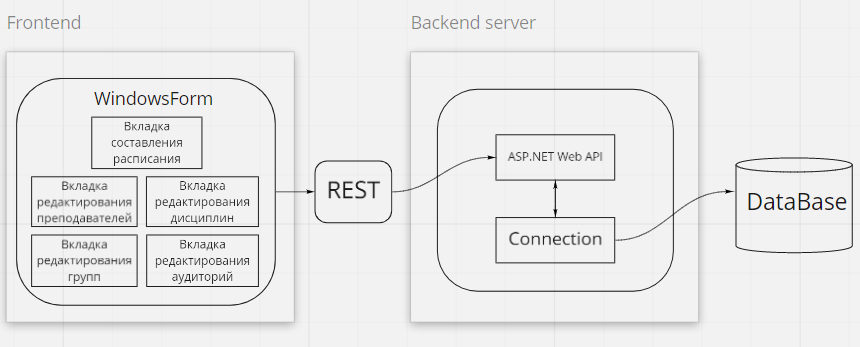
Вся работа по разработке системы составления расписания должна быть документирована в соответствии со стандартами.

Документирование проекта в рамках Технического Задания ведется в соответствии с ГОСТ 34. Данный документ является основой для анализа предметной области.

Документирование тестирования должно быть представлено в виде электронной таблицы отдельным файлом.

# **Приложение**

Ниже представлена модульная схема разрабатываемого приложения.



Пользователь будет взаимодействовать с приложением WindowsForm.Соответственно, frontend - это форма приложения,которая состоит их 5 вкладок:

* Вкладка составления расписания – на этой вкладке будет таблица существующего расписания, либо чистая таблица, для создания нового. Будет возможность добавления или удаления занятий для определенной группы в определенный день. Занятия выбираются из выпадающего списка с преподавателями, аудиториями и дисциплинами.
* Вкладка редактирования преподавателей – здесь будет таблица со списком преподавателей на факультете и будет возможность просмотреть список предметов для каждого преподавателя, так же можно будет вносить изменения в эти списки. Здесь же происходит обработка ввода(пользователь не сможет ввести цифры и иные символы в поля для ФИО преподавателя кроме символом русского и английского алфавита)
* Вкладка редактирования дисциплин – данная вкладка показывает дисциплины, записанные в базе данных. Пользователь также может внести в него свои изменения(добавить, удалить, изменить дисциплину).Так же есть ограничение на ввод, в поле можно ввести только символы русского или английского алфавита.
* Вкладка редактирования групп – на данной вкладке пользователь может увидеть список групп и может его редактировать.
* Вкладка редактирования аудиторий – здесь пользователь может увидеть доступные для факультета аудитории и аналогично может редактировать данный список.

Взаимодействие между frontend происходит посредством REST. То есть все взаимодействие с сервером сводится к 4 операциям :

* Получение данных с сервера, для которого используется метод HTTP-запроса GET.
* Добавление новых данных на сервер, для которого используется метод HTTP-запроса POST.
* Модификация существующих данных на сервере, для которого используется метод HTTP-запроса PUT.
* Удаление данных на сервере, для которого используется метод HTTP-запроса DELETE.

Контроллеры в ASP.NET Web API принимают запрос, обрабатывает его и отправляет ответ. В Connection хранится соединения с базой данных.