**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информатика»

Специальность 1-40 04 01 «Информатика и технологии программирования»

**Отчёт руководителя проекта** **«Нагрузка кафедры»**

по дисциплине: **«Управление разработкой программного обеспечения»**

Руководитель: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Исполнители: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Заказчик: Шибеко В.Н.

Дата проверки: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Гомель 2021

**Содержание**

[Введение 3](#_Toc65750667)

[1 ОБЗОР МЕТОДОЛОГИИ РАЗРАБОТКИ ПРОЕКТА 4](#_Toc65750668)

[1.1 Состав команды 4](#_Toc65750669)

[1.2 Методология и способы коммуникации 4](#_Toc65750670)

[2 ТРЕБОВАНИЯ ПРОЕКТА 5](#_Toc65750671)

[2.1 Функциональные требования 5](#_Toc65750672)

[2.2 Диаграмма прецедентов 5](#_Toc65750673)

[2.3 Журнал проекта 6](#_Toc65750674)

[3 ПЛАНИРОВАНИЕ 7](#_Toc65750675)

[3.1 Журнал релиза 7](#_Toc65750676)

[3.2 Календарный план-график 7](#_Toc65750677)

[3.3 Список тестов 9](#_Toc65750678)

[3.4 Управление версиями программного продукта 9](#_Toc65750679)

[4 МОНИТОРИНГ, АНАЛИЗ И ПОДГОТОВКА К НОВОМУ РЕЛИЗУ 10](#_Toc65750680)

[4.1 Мониторинг 10](#_Toc65750681)

[4.2 Анализ готовности продукта 11](#_Toc65750682)

[4.3 Подготовка к новому релизу 14](#_Toc65750683)

[Заключение 16](#_Toc65750684)

[Список использованных источников 17](#_Toc65750685)

# **Введение**

Одной из актуальных проблем работы кафедр учебного заведения является распределение и учет выполнения учебной нагрузки преподавательского состава. Выполнение этой работы занимает много времени, неизбежны ошибки и многочисленные корректировки.

Проблемой автоматизации управления своими подразделениями занимаются многие ВУЗы. К сожалению, несмотря на позитивные сдвиги, наметившиеся в последние годы, распределение и учет выполнения учебной нагрузки до сих пор выполняется вручную.

Для сокращения непродуктивных потерь времени при распределении и учете выполнения нагрузки предлагается автоматизировать процесс при помощи программного обеспечения. Это позволит существенно сократить время, затрачиваемое на распределение учебных часов преподавателям кафедры в сравнении с используемыми в настоящее время способами, даст возможность исключить ошибки, а также, в случае необходимости, оперативно внести поправки в план учебной нагрузки [4].

На распределение нагрузки влияют следующие факторы:

* доля участия каждого преподавателя в работе кафедры (штатная единица);
* количество преподавателей, ведущих занятия в учебной группе;
* количество студентов в учебной группе;
* наличие лекционных часов заданного потока у конкретного преподавателя;
* количество учебных групп заданного потока у конкретного преподавателя.

Целью проекта является разработка единого приложения, которое позволит объединить два приложения ­­­­– по подготовке индивидуального плана на год и на семестр­ – путем обращения к ним.

# **1 ОБЗОР МЕТОДОЛОГИИ РАЗРАБОТКИ ПРОЕКТА**

## **1.1 Состав команды**

Руководитель проекта: Процкая Мария.

Команда проекта:

* Процкая Мария – менеджер проекта, разработчик, тестировщик;
* Архипенко Мария – разработчик, тестировщик;
* Земченок Вадим – разработчик, тестировщик.

Основные задачи:

Менеджер проекта – общение с заказчиком, согласование целей и задач проекта, подготовка требований проекта, подготовка структуры проекта, подготовка основных диаграмм, подготовка документации проекта.

Разработчик – разработка интерфейса, разработка программных частей проекта, выпуск хот-фиксов и баг-фиксов.

Тестировщик – проверка работоспособности проекта, выявления ошибок и недочётов, проверка корректности работы проекта.

**1.2 Методология и способы коммуникации**

Для разработки проекта была выбрана методология *Kanban*. Преимущества данной методологии: гибкость планирования, высокое вовлечение команды в процесс разработки, наглядность.

В качестве средства коммуникации был выбран мессенджер *Telegram*. Данное средство коммуникации было выбрано среди множества других, так как имеет ряд преимуществ, такие как: групповые чаты, удобный интерфейс, скорость и безопасность, хранение всех файлов в облаке и, как следствие, доступность со всех устройств.

# **2 ТРЕБОВАНИЯ ПРОЕКТА**

## **2.1 Функциональные требования**

Необходимо разработать интеграционное приложение, включающее обращение к двум приложениям по вычислению нагрузки, выгрузку данных в базу данных *MS Access*, вести контроль хода разработки.

Процессы для разработки:

1. Обращение к первому приложению.
2. Обращение ко второму приложению.
3. Обращение к веб-порталу.
4. Запуск *jar-*файла с последующей загрузкой данных.
5. Выгрузка данных в базу данных.

Лица, участвующие в процессах:

1. Лаборант кафедры.

## **2.2 Диаграмма прецедентов**

Для проекта была разработана диаграмма прецедентов, представленная на рисунке 2.1. Данная диаграмма отображает роли и последовательность действий в данном проекте.

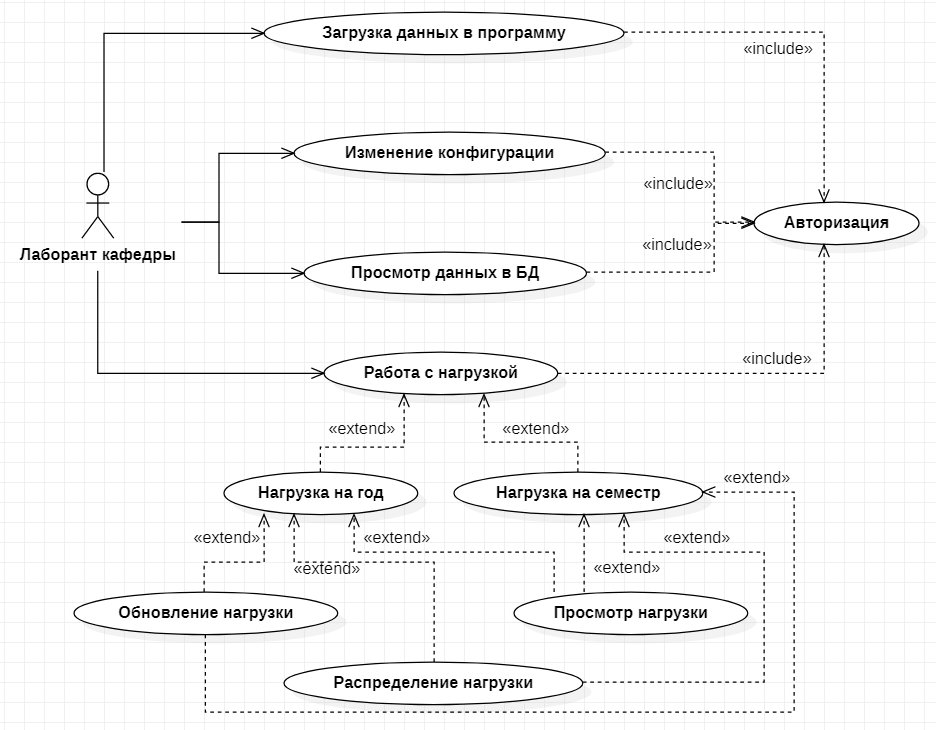


Рисунок 2.1 – Диаграмма прецедентов

## **2.3 Журнал проекта**

Как лаборант кафедры, я хочу иметь возможность просмотреть данные в базе данных, а также открыть отдельные окна-приложения для последующей работы с нагрузкой на год и на семестр.

# **3 ПЛАНИРОВАНИЕ**

## **3.1 Журнал релиза**

Срок разработки проекта составил 5 недель. Было определено 4 итерации разработки.

Первая итерация:

* сформировать требования заказчика;
* составить журнал проекта, распределить роли в команде;
* создать *Use Case* диаграмму.

Вторая итерация:

* реализовать интеграционное приложение;
* тестирование приложения.

Третья итерация:

* реализовать взаимодействие между приложениями;
* тестирование правильности работы приложения.

Четвертая итерация:

* тестирование всего интеграционного приложения в целом;
* заполнение документации проекта.

## **3.2 Календарный план-график**

Для наглядности отчета о выполнении каждого этапа был составлен календарный план-график, представленный в таблице 3.1

Таблица 3.1. Календарный план-график

|  |  |  |  |
| --- | --- | --- | --- |
| Задача | Дата выдачи задачи | Исполнитель | Отметка о выполнении |
| Определение обязанностей каждого члена команды | 28.01.2021 | Процкая М. А. | + |
| Создание репозитория на *GitHub* | 10.02.2021 | Процкая М. А. | + |
| Создание доски *Kanban* | 10.02.2021 | Процкая М. А. | + |
| Создание интеграционного приложения | 10.02.2021 | Процкая М. А. | + |
| Авторизация и аутентификация | 10.02.2021 | Процкая М. А., Земченок В. Г. | + |
| Обеспечение вызова приложений из интеграционного, связь с ними | 10.02.2021 | Процкая М. А., Земченок В. Г., Архипенко М. А. | + |
| Запуск *jar* файла | 19.02.2021 | Земченок В. Г. | + |
| Настройки конфигурации для выбора папки | 19.02.2021 | Архипенко М. А. | + |
| Экспорт данных из *JSON* в базу данных *MS* *Access* | 10.02.2021 | Земченок В. Г., Архипенко М. А. | + |
| Отчет | 18.02.2021 | Процкая М. А. | + |
| Презентация | 25.02.2021 | Процкая М. А. | + |
| Сдача проекта | 05.03.2021 | Процкая М. А., Земченок В. Г.,  Архипенко М. А. | + |
| Защита проекта | 05.03.2021 | Процкая М. А., Земченок В. Г.,  Архипенко М. А. | + |

## **3.3 Список тестов**

Ввиду отсутствия какой-либо сложной логики, требующей детального тестирования, было принято решение не проводить *Unit*-тестирование программы.

Тесты, которые будут проведены:

* авторизация (валидация полей, корректность входа);
* запуск двух частей программы;
* корректность загрузки данных из *JSON* в *MS* *Access*;
* изменение конфигурации;
* тестирование удобства использования.

## **3.4 Управление версиями программного продукта**

Управление версиями в проекте происходит при помощи синхронизации с *GitHub*, в котором также происходит планирование с помощью доски *Kanban*. Подробная инструкция по интеграции *Git* и *Visual Studio* представлена в источниках [1] и [2].

# **4 МОНИТОРИНГ, АНАЛИЗ И ПОДГОТОВКА К НОВОМУ РЕЛИЗУ**

## **4.1 Мониторинг**

На рисунках 4.1-4.2 представлена доска планирования задач *Kanban*.

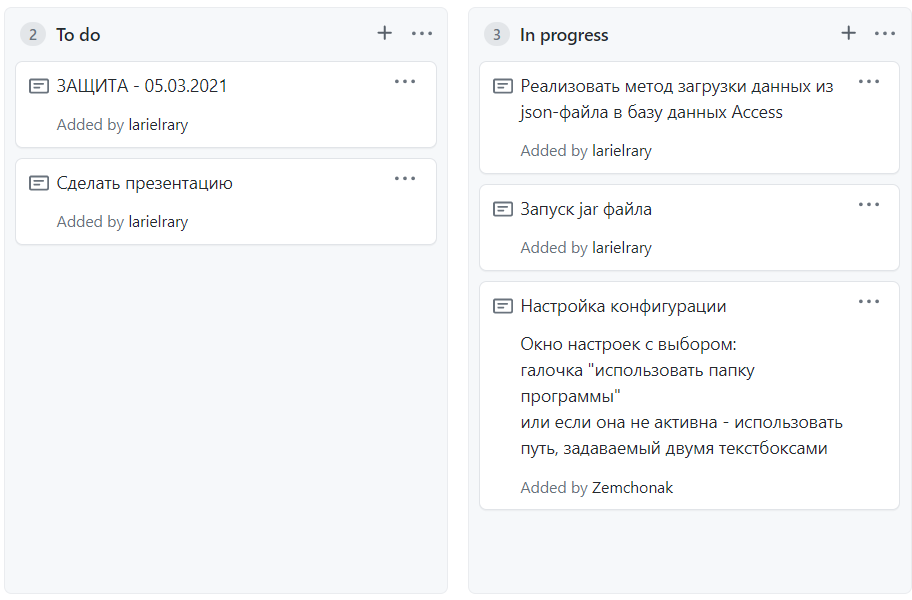


Рисунок 4.1 – Доска планирования задач *Kanban*, этапы *To do* и *In progress*

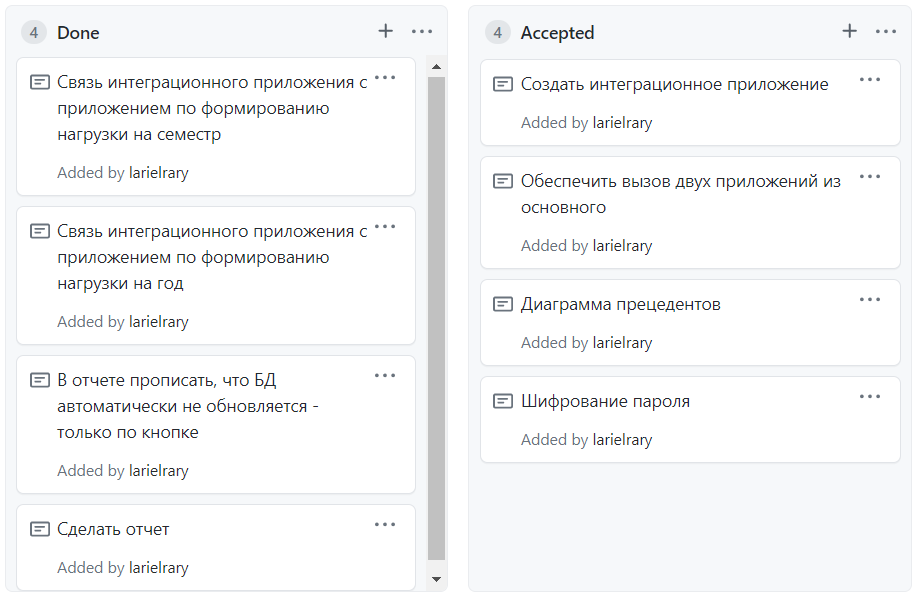


Рисунок 4.2 – Доска планирования задач Kanban, этапы *Done* и *Accepted*

## **4.2 Анализ готовности продукта**

После завершения разработки проекта был проведен анализ готовности продукта согласно план-графику.

Все запланированные этапы разработки были выполнены.

Проект был протестирован.

Если при авторизации пользователь вводит неверные данные, он получает соответствующее сообщение с предложением повторить попытку. Скриншот сообщения представлен на рисунке 4.3.

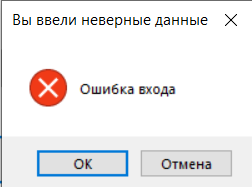


Рисунок 4.3 – Уведомление о вводе неверных данных при авторизации

Если при нажатии кнопки экспорта данных происходит ошибка (например, отсутствует *MS Access*), пользователь получает соответствующее сообщение с предложением повторить попытку. Скриншот сообщения представлен на рисунке 4.4.

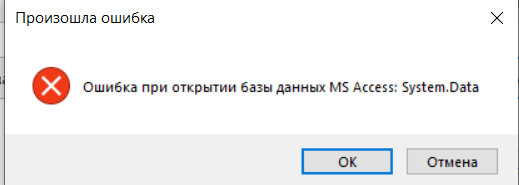


Рисунок 4.4 – Уведомление об ошибке при экспорте в *Access*

Если при нажатии кнопки загрузки данных происходит ошибка (например, на компьютере не установлен пакет *Java*), пользователь получает соответствующее сообщение с предложением повторить попытку. Скриншот сообщения представлен на рисунке 4.5.

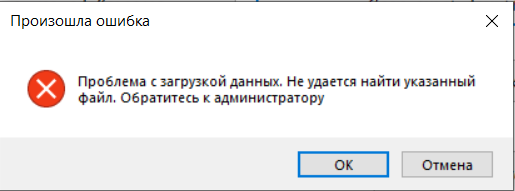


Рисунок 4.5 – Уведомление об ошибке при запуске *jar*-файла

Основные тесты интерфейса представлены в таблице 4.1

Таблица 4.1 – Тестирование пользовательского интерфейса.

|  |  |  |
| --- | --- | --- |
| **№** | **Вид проверки** | **Результат** |
| 1 | Реализуется ли функционал приложения при помощи графических элементов | 10 |
| 2 | Реализуется ли размещение сообщений об ошибках | 10 |
| 3 | Отсутствуют ли орфографически, пунктуационные ошибки | 10 |
| 4 | Читабелен ли используемый шрифт | 9 |

Результат тестирования функционала программы представлен в таблице 4.2.

Таблица 4.2 – Тестирование функционала

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Имя элемента** | **Тест** | **Результат** |
| 1 | *YearBtnClick(), TeacherBtnClick()* | Запуск двух частей программы | Исключительных ситуаций нет |
| 2 | *ExportToAccessBtnClick()* | Загрузка данных из *JSON* в *MS Access* | Исключительных ситуаций нет |
| 3 | *ConfigurationBtnClick()* | Изменение конфигурации | Исключительных ситуаций нет |
| 4 | *LoadJarBtnClick()* | Запуск *JAR*-файла | Исключительных ситуаций нет |
| 5 | *CheckInputData()* | Авторизация | Исключительных ситуаций нет |
| 6 | *Encryption()* | Шифрование пароля | Исключительных ситуаций нет |

## **4.3 Подготовка к новому релизу**

Все предложения участников проекта учтены и записаны в журнал проекта. После завершения всех требуемых итераций разработки, заказчику демонстрируется демоверсия проекта (на рисунках 4.6 – 4.8 представлены скриншоты приложения), в котором учтены замечания и предложения. Проект полностью готов к использованию, возможны его дальнейшие усовершенствования, такие как: автоматическое обновление данных в базе данных при обновлении *JSON*-файла и т.д.

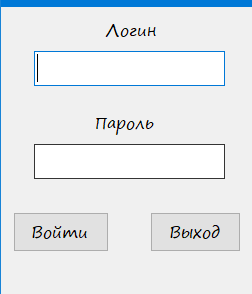
****

Рисунок 4.6 – Форма авторизации

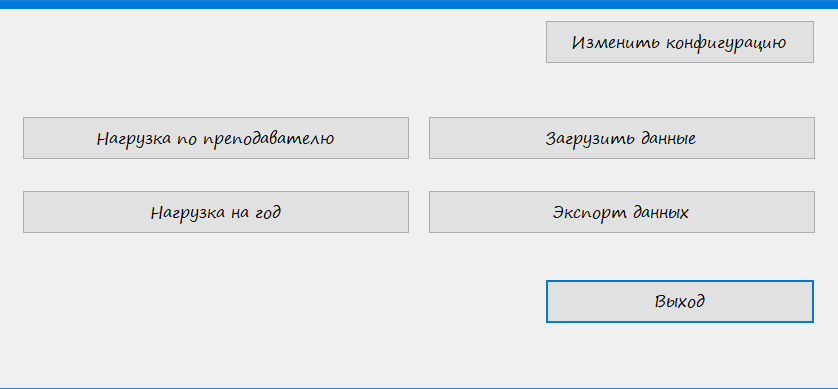


Рисунок 4.7 – Главная форма

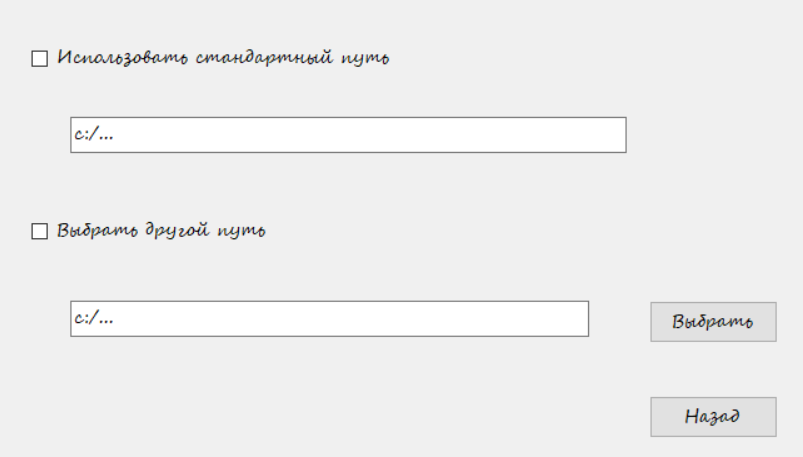


Рисунок 4.8 – Форма изменения конфигурации

# **Заключение**

Поставленная задача решена на языке программирования *C*# [3] в среде разработки *Microsoft Visual Studio Community* 2019.

Рассмотренная технология автоматизированного распределения и учета выполнения учебной нагрузки значительно сокращает время, затрачиваемое на распределение учебных часов преподавателям кафедры, позволяет исключить ошибки, а также, в случае необходимости, быстро вносить поправки в план учебной нагрузки.

Поставленную задачу можно считать полностью выполненной.

# **Список использованных источников**

1. *Git, GitHub* и *Visual Studio.* В помощь новичкам [Электронные ресурс].: Хабр. – Электронные данные. – Режим доступа: https://habr.com/ru/sandbox /112936/ – Дата доступа: 17.02.2021.
2. *GitHub* для пользователей *Windows* [Электронные ресурс].: Хабр. – Электронные данные. – Режим доступа: https://habr.com/ru/post/313996/ – Дата доступа: 17.02.2021.
3. Рихтер. Программирование на платформе Microsoft .NET/ В. Л. Григорьев. – Москва: Энергоатомиздат, 2010. – 209 с.
4. Автоматизация процесса распределения и учета учебной нагрузки [Электронные ресурс].: Киберленинка. – Электронные данные. – Режим доступа: https://cyberleninka.ru/article/n/avtomatizatsiya-protsessa-raspredeleniya-i-ucheta-uchebnoy-nagruzki-prepodavatelya/viewer – Дата доступа: 02.03.2021.