Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа: Инженерная школа энергетики

Направление: Прикладная информатика

Отделение: Электроэнергетики и электротехники

**Отчет по лабораторной работе №6**

«Проектная документация»

по дисциплине: «Основы объектно-ориентированного программирование»

|  |  |  |
| --- | --- | --- |
| Выполнил студент гр. О-5КМ91: |  | Грищенко Л.Д. |
|  | (Дата, подпись) | (Ф. И. О.) |

|  |  |  |
| --- | --- | --- |
| Проверил: к.т.н., доцент |  | Калентьев А.А. |
| (Степень, звание, должность) | (Дата, подпись) | (Ф. И. О.) |

Томск – 2021 г.

**СОДЕРЖАНИЕ**

[1 Техническое задание 3](#_Toc74224276)

[1.1 Предпроектный анализ 3](#_Toc74224277)

[ВВЕДЕНИЕ 4](#_Toc74224278)

[2 Основная часть 5](#_Toc74224279)

[3 Тестирование 8](#_Toc74224280)

[ЗАКЛЮЧЕНИЕ 9](#_Toc74224281)

[СПИСОК ИСПОЛЬЗОВАННЫХ СИТОЧНИКОВ 10](#_Toc74224282)

[Приложение А Техническое задание 11](#_Toc74224283)

# Техническое задание

## Предпроектный анализ

Проблема: у индивидуального предпринимателя нет удобной программы для начисления заработной платы работникам различными способами начисления.

Цель: разработать быструю программу для начисления заработной платы работникам различными способами начисления.

Задачи:

* Программа должна предоставлять следующие способы начисления зарплаты: почасовая оплата, оплата по окладу и ставке
* Программа должна предоставлять возможность отображения работников, которым была начислена зарплата.
* Программа должна иметь возможность сохранять список сотрудников с начисленной им заработной платой в специальном формате.
* Программа должна иметь возможность редактировать список сотрудников с начисленной им заработной платой.

Техническое задание приведено в приложении А.

# ВВЕДЕНИЕ

Цель: создание проектной документации по разработанной программе.

Программная документация предназначена для сопровождения разработанного программного средства

В рамках лабораторной работы необходимо проделать следующее

* разработать техническое задание;
* составить UML диаграмма вариантов использования;
* составить UML диаграмма классов;
* для классов, образующих связь типа «общее-частное» (наследование, реализация), привести описание в виде таблиц;
* привести дерево ветвления Git;
* произвести тестирование программы.

# Основная часть

На рисунке 2.1 приведена UML диаграмма вариантов использования для разработанной программы [1].

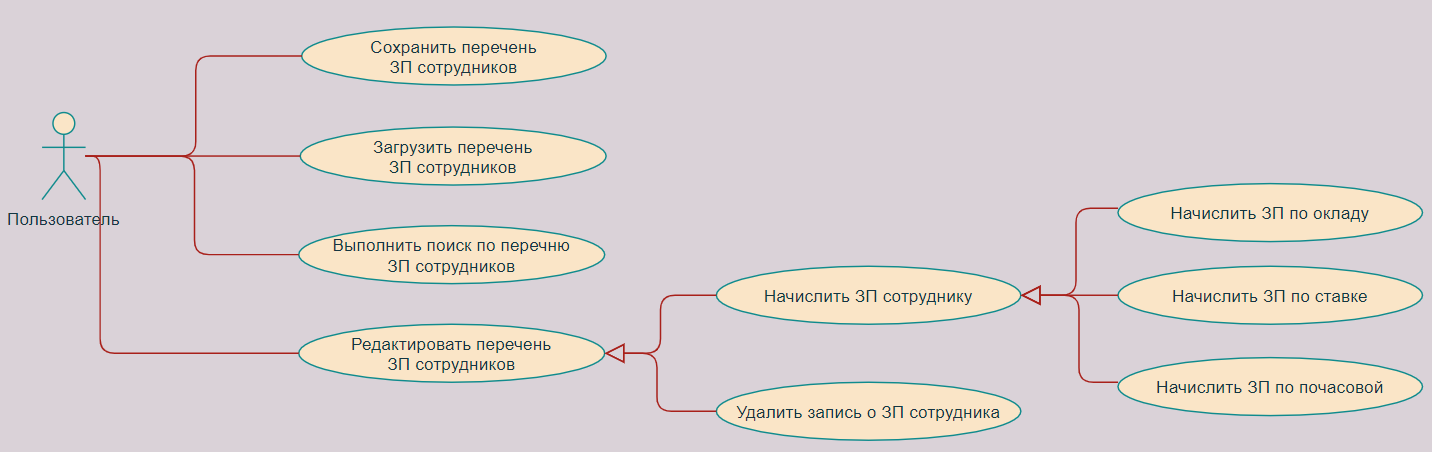
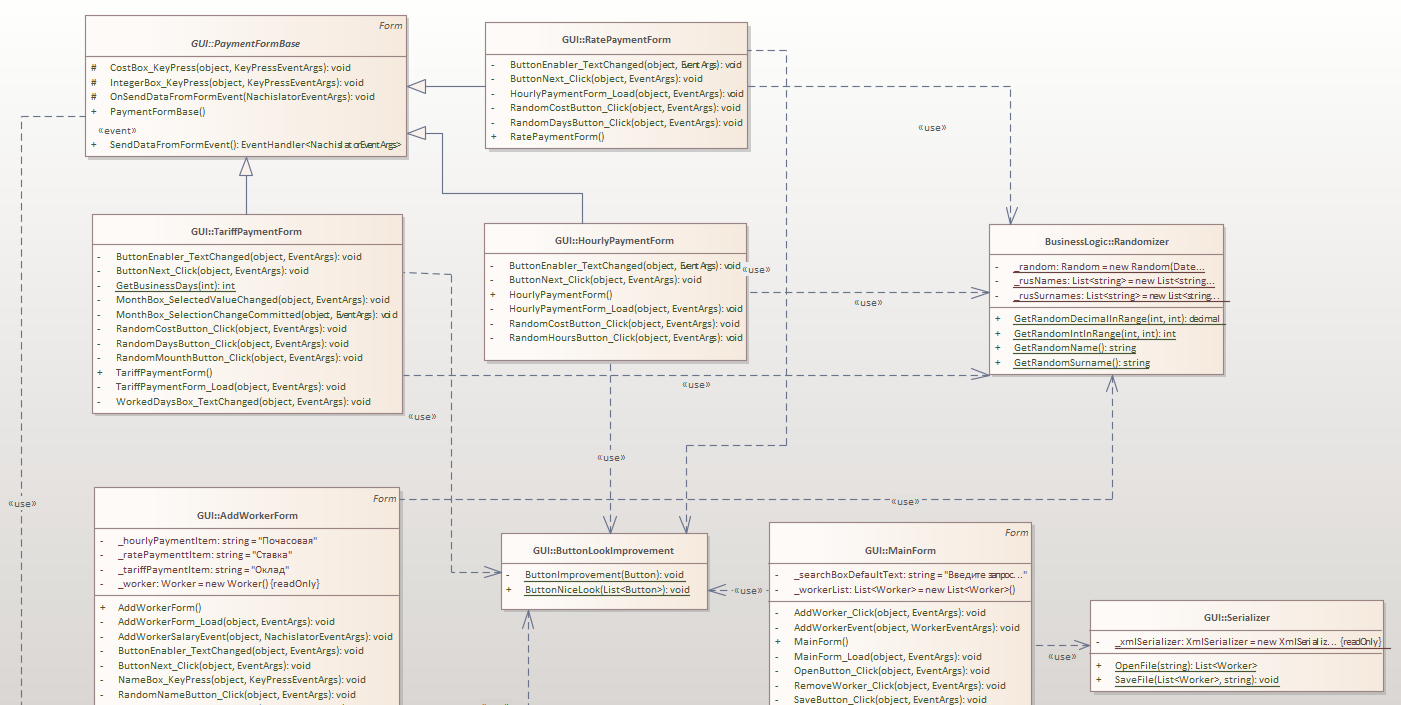


Рисунок 2.1 – UML диаграмма вариантов использования

На рисунке 2.2 приведена UML диаграмма классов [1].



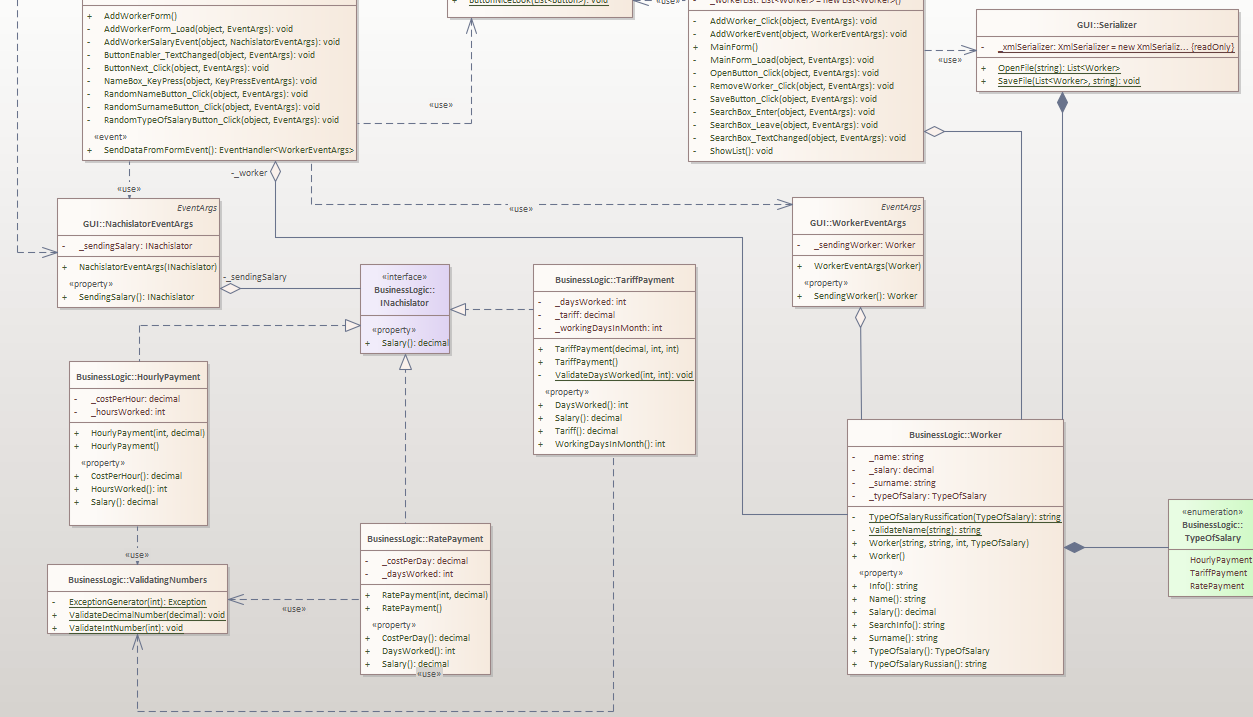


Рисунок 2.2 – UML диаграмма классов

Для классов, образующих связь типа «общее-частное» (наследование, реализация), ниже приведено описание (таблицы 2.1 – 2.4).

Таблица 2.1 – Описание интерфейса INachslator

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| Описание класса | | |
| Интерфейс INachslator – интерфейс для начисления ЗП | | |
| Свойства | | |
| +Salary | decimal | Зарплата |

Таблица 2.2 – Описание класса HourlyPayment

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| Описание класса | | |
| HourlyPayment – Класс для почасовой оплаты | | |
| Свойства | | |
| + HoursWorked | int | Отработанные часы |
| + CostPerHour | decimal | Стоимость часа |
| +Salary | decimal | Зарплата |
| Методы | | |
| + HourlyPayment(int hoursWorked, decimal costPerHour) |  | Конструктор для создания нового экземпляра почасовой оплаты |

Таблица 2.3 – Описание класса RatePayment

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| Описание класса | | |
| RatePayment – Класс для оплаты по ставке | | |
| Свойства | | |
| + DaysWorked | int | Отработанные дни |
| + CostPerDay | decimal | Стоимость дня |
| +Salary | decimal | Зарплата |
| Методы | | |
| + RatePayment(int daysWorked, decimal costPerDay) |  | Конструктор для создания нового экземпляра оплаты по ставке |

Таблица 2.4 – Описание класса TariffPayment

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| Описание класса | | |
| TariffPayment – Класс для оплаты по окладу | | |
| Свойства | | |
| + DaysWorked | int | Отработанные дни |
| + WorkingDaysInMonth | int | Количество рабочих дней в месяце |
| + Tariff | decimal | Оклад |
| +Salary | decimal | Зарплата |
| Методы | | |
| + TariffPayment(decimal tariff, int workingDaysInMonth,int daysWorked) |  | Конструктор для создания нового экземпляра оплаты по окладу |
| - ValidateDaysWorked(int daysWorked, int workingDaysInMonth) | void | Проверка ввода количества дней |

На рисунке 2.3 приведено дерево ветвления Git, полученной по окончанию работы с проектом.

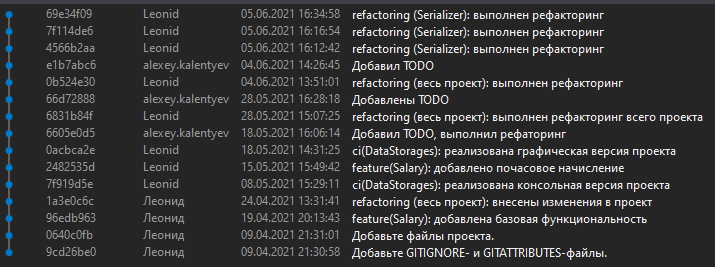


Рисунок 2.3 – Дерево ветвления Git

# Тестирование

В рамках функционального тестирования была вывалена ошибка, в результате которой при попытке сгенерировать случайную величину на форме TariffPaymentForm программа прекращала свою работу. Для решения данной ошибки было решено сделать кнопку генерации случайной величины не активной до тех пор, пока не будет заполнено соответствующее поле.

# ЗАКЛЮЧЕНИЕ

В результате выполнения лабораторной работы было сделано следующее:

* разработано техническое задание;
* составлена UML диаграмма вариантов использования;
* составлена UML диаграмма классов;
* для классов, образующих связь типа «общее-частное» (наследование, реализация), приведено описание в виде таблиц;
* приведено дерево ветвления Git;
* произведено тестирование программы.

# СПИСОК ИСПОЛЬЗОВАННЫХ СИТОЧНИКОВ

1. Калентьев А. А. Новые технологии в программировании : учеб. пособие / А. А. Калентьев, Д. В. Гарайс, А. Е. Горяинов. — Томск : Эль Контент, 2014. — 176 с.

2. Трудовой кодекс Российской Федерации

# Приложение А Техническое задание

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

на разработку программного средства по теме

«Создание программного средства для начисления заработной платы»

# ОБЩИЕ ПОЛОЖЕНИЯ

## Полное наименование системы и ее условное обозначение

Полное наименование: «Программное обеспечение для начисления заработной платы сотрудникам различными способами: почасовая оплата, оплата по окладу и оплата по ставке».

Краткое наименование: «Начисление заработной платы».

Условное обозначение: «Система».

## Сведения о заказчике и исполнителе

Заказчик: Индивидуальный предприниматель Калентьев Алексей Анатольевич.

Исполнитель: студент Томского политехнического университета Грищенко Леонид Дмитриевич.

## Перечень сокращений

|  |  |
| --- | --- |
| ЗП | Заработная плата |

# Назначение и цели создания системы

## Назначение Системы

Разрабатываемая система предназначена для начисления ЗП сотрудникам различными способами: почасовая оплата, оплата по окладу и оплата по ставке.

## Цели создания Системы

Разрабатываемая система создается с целью уменьшения трудозатрат заказчика, а также для автоматизации процесса начисления ЗП.

# Характеристика объектов автоматизации

Процесс расчета ЗП выполняется в соответствии с требованиями законодательства и не является индивидуальным для каждого работника, а меняется только в зависимости от способа начисления. Исходя из этого процесс начисления ЗП может быть автоматизирован.

# Требования к Системе

## Требования к структуре Системы

**A01.** Система должна являться настольным приложением.

## Требования к функциям Системы

**F01.** Должна быть реализована возможность начисления ЗП следующими способами: почасовая оплата, оплата по окладу и оплата по ставке

**F01.1.** Для начисления ЗП по почасовой оплате должны задаваться количество отработанных часов и стоимость одного часа.

**F01.2.** Для начисления ЗП по окладу должны задаваться номер месяца, величина оклада и количество отработанных дней.

**F01.3** Для начисления ЗП по ставке должны задаваться количество дней, размер ставки.

**F02.** Расчет ЗП должен производиться в соответствии с нормативными документами [2]

**D01.** Должна быть возможность сохранения данных в формате GLD.

**D02.** Должна быть возможность загрузки данных в формате GLD.

**S01.** Установка и функционирование Системы не должны приводить к необходимости изменения настроек антивирусного программного обеспечения.

**U01.** Система должна иметь графический интерфейс пользователя.

**U02.** Должна быть реализована система обработки ошибок.

## Требования к видам обеспечения

Требования к программному обеспечению.

**C01.** Система должна работать с операционной системой Windows 10 Professional Version 2004.

**C02.** Система должна работать с платформой .NET Framework 5.0.

Требования к аппаратному обеспечению.

**C03.1** Процессор Intel Pentium 4 или более поздней версии с поддержкой SSE3

**C03.2** Объем оперативной памяти – 2 ГБ.

**C03.3** Количество свободного места на диске – 50 МБ