Министерство науки и высшего образования РФ

Федеральное государственное образовательное учреждение

высшего образования

«УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»

Кафедра АСУ

Курсовая работа

по дисциплине

«Объектно-ориентированное моделирование

и программирование»

«Совместное написание программного обеспечения»

Выполнили:

Студенты гр. ПИ-101Бзу

Поленов И.C.

Тохтаров Д.И.

Проверил:

преподаватель

Казанцев А. В.

г. Уфа 2024

**Министерство науки высшего образования РФ**

Федеральное государственное образовательное учреждение

высшего образования

«УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»

Кафедра автоматизированных систем управления

ЗАДАНИЕ

на курсовую работу (КР) по дисциплине «Объектно-ориентированное моделирование и программирование»

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Студент | *Поленов И.C.* | | Группа | *ПИ-101Бзу* | Консультант | *Казанцев А.В.* |
|  | Фамилия И.О. | |  | номер группы |  | Фамилия И.О. |
| Студент | *Тохтаров Д.И.* | | Группа | *ПИ-101Бзу* | Консультант | *Казанцев А.В.* |
|  | Фамилия И.О. | |  | номер группы |  | Фамилия И.О. |
|  |  | |  |  |  |  |
| 1.Тема КР: | | Зарплатный калькулятор. Повременная форма оплаты труда | | | | | |
|  | | наименование темы | | | | | |

2.Основное содержание

1. Пояснительная записка с необходимыми материалами.

2. Репозиторий системы контроля версий содержащий программный код с комментариями.

3. Требования к оформлению:

3.1. Пояснительная записка должна быть оформлена в текстовом процессоре Office Word в соответствии с требованиями ГОСТ. В бумажном виде оформляется: титульный лист, бланк задания, также прикладываются аннотации, где указываются ссылка на репозиторий, содержащий исходный проект, работающую программу и пояснительную записку.

4. Графическая часть должна включать:

- экранные формы инструментальных средств;

- экранные формы, разрабатываемого программного продукта.

Дата выдачи 02.12.2023 Дата окончания 28.03.2024

Руководитель *Казанцев А.В.* Студент *Поленов И.C*

Студент Тохтаров Д.И.

**Содержание**

1. Описание предметной области..................................................................................4

2. Описание прототипа ПП............................................................................................6

3. Математическая модель работы ПП.........................................................................8

4. Написание программного кода по зонам ответственности....................................9

5. Тестирование ПП......................................................................................................12

Заключение....................................................................................................................13

Приложение П-1............................................................................................................14

Приложение П-2............................................................................................................15

Список использованной литературы..............................................................................

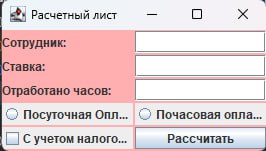
1. Описание предметной области

Любой труд должен достойно оплачиваться, а основную составляющую любой себестоимости продукции составляет именно зарплата сотрудникам любой компании и производственных предприятий. ТВ теории по экономике выделяют следующие концепции определения зарплаты:   
 1) Заработная плата – это стоимость труда. В свою очередь ее размер складывается под влиянием рыночных факторов, к примеру, от предложения на рынке или его спросе; 2) Заработная плата – это денежное выражение стоимости «рабочей силы». Ее размер складывается от производства продукции и от факторов, из-за которых возникает разница между заработной платы и стоимости рабочей силы сотрудников.   
Оплата труда – это средства, выплачиваемые всему коллективу предприятия, в денежной форме, за выполненную рабочую деятельность за определенное время и учитывающее законодательство страны.  
 В данном курсовом проекте был реализован зарплатный калькулятор повременной оплатой труда, основными элементами предметной области являются:

1. Тарифные ставки: это информация о ставках оплаты труда, которые применяются к различным категориям работников в организации. Тарифные ставки могут быть установлены в зависимости от уровня квалификации, должности, стажа работы и других факторов. В зарплатном калькуляторе должна быть возможность задания и хранения различных тарифных ставок для каждой категории работников.  
 2. Рабочее время: это учет отработанных часов или дней сотрудниками. Зарплатный калькулятор должен позволять вводить данные о рабочем времени сотрудника, а также учитывать переработки, недоработки, отпуска и больничные. Также важно иметь возможность автоматического расчета рабочего времени на основе введенных данных.  
 3. Дополнительные выплаты: кроме основной заработной платы, сотрудники могут получать различные дополнительные выплаты, такие как премии, надбавки за работу в ночное время, выходные дни и другие. Зарплатный калькулятор должен учитывать эти дополнительные выплаты и автоматически добавлять их к общей сумме начислений.  
 4. Налоги и удержания: при начислении заработной платы необходимо учитывать налоги и другие обязательные удержания. Зарплатный калькулятор должен автоматически рассчитывать налоги на доходы физических лиц, взносы в ПФР и ФСС, а также другие обязательные платежи и удержания.  
 5. Интеграция с другими системами: для эффективного управления персоналом и бухгалтерского учета важно иметь возможность интеграции зарплатного калькулятора с другими информационными системами компании, такими как бухгалтерские программы, системы управления персоналом и другие. Это позволит автоматизировать обмен данными между системами и упростить процессы управления персоналом и финансами.

2 Описание прототипа ПП

Зарплатный калькулятор - это инструмент, который позволяет рассчитать заработную плату сотрудников на основе различных параметров, таких как оклад, отработанные часы, премии, налоги и другие факторы. Этот инструмент обычно используется для удобного и быстрого расчета заработной платы сотрудников, а также для оценки различных сценариев оплаты труда.Зарплатный калькулятор может включать функции, позволяющие учитывать различные налоговые вычеты, страховые взносы, дополнительные выплаты и другие финансовые аспекты, влияющие на конечную сумму, которую получает сотрудник на руки. Такие калькуляторы обычно предоставляют возможность рассчитывать как чистую сумму заработной платы, так и общие затраты работодателя на выплату заработной платы. Использование зарплатного калькулятора помогает как работодателям, так и сотрудникам более прозрачно понимать структуру и размер заработной платы, а также облегчает процесс планирования бюджета и учета финансовых обязательств по оплате труда.

На экранной форме зарплатного калькулятора были представлены:  
Рис.1 Экранная форма.

1. Текстовое поле для ввода имени сотрудника. (JLabel)
2. Числовое поле для ввода ставки. (JLabel)
3. Радиокнопки посуточной/почасовой оплаты. (JRadioButton)
4. Чекбокс для выбора с учетом налоговой ставки или нет. (JCheckBox)
5. Кнопка для начала расчета. (JButton)
6. Контекстное меню для вывода информации. (JFrame)
7. Кнопка “ок” для закрытия контекстного меню. (JButton)
8. Контекстные меню “ошибка”. (JFrame)

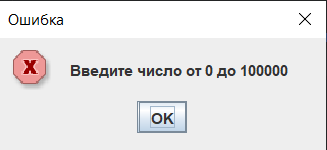


Рис.2 Контекстное меню Ошибки 1

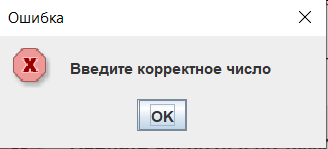


Рис.3 Контекстное меню Ошибки 2

3 Математическая модель работы ПП

В математической модели работы программы” Фитнес калькулятор” заложено на выполнение двух функций это: расчет индекса массы тела и расчет частоты сердечного сокращения пульсовых зон

В математической модели программы зарплатного калькулятора можно использовать различные формулы для расчета заработной платы, налогов, и других финансовых параметров. Ниже приведены основные формулы, которые использовались при разработке зарплатного калькулятора:

1. Расчет заработной платы по тарифной ставке:  
 Заработная плата = Тарифная ставка \* Отработанные часы/Отработанные дни  
2. Расчет налога на доходы физических лиц (НДФЛ):  
 - НДФЛ = Заработная плата \* Ставка НДФЛ  
3.Расчет общей суммы к выплате:  
- Общая сумма к выплате = Заработная плата - НДФЛ

**4 Написание программного кода по зонам ответственности**

Зона ответственности определяет, кто из участников является исполнителем той или иной задачи при написании программного кода и тестировании.

Разработка ПО ведется на языке высокого уровня Java с использованием инструментальной среды IntelliJ IDEA 2023.3.4.

В таблице 1 присутствует список разработчиков, а также их зоны ответственности.

Таблица 1 – Зоны ответственности

|  |  |  |
| --- | --- | --- |
| **№** | **ФИО разработчика/модератора** | **Зона ответственности** |
| 1 | Поленов И. С. | Реализация GUI метода, создание проверки для вводимого текста, проверка работоспособности кода и редактирование недостатков |
| 2 | Тохтаров Д. И. | Реализация класса ButtonEventListener, реализация String message для отображения результата |

Таблица 1 – Зоны ответственности

Ссылка на репозиторий проекта: https://github.com/IvanPolenov/ZP\_Calc.git

Инструкция по подключению к Git:

Шаг 1: Установка плагина Git  
- Откройте IntelliJ IDEA и выберите "File" -> "Settings" (или "IntelliJ IDEA" -> "Preferences" на macOS)  
- В разделе "Plugins" найдите плагин Git и убедитесь, что он установлен. Если плагин не установлен, нажмите на кнопку "Browse repositories" и найдите Git. Установите плагин и перезапустите IntelliJ IDEA.

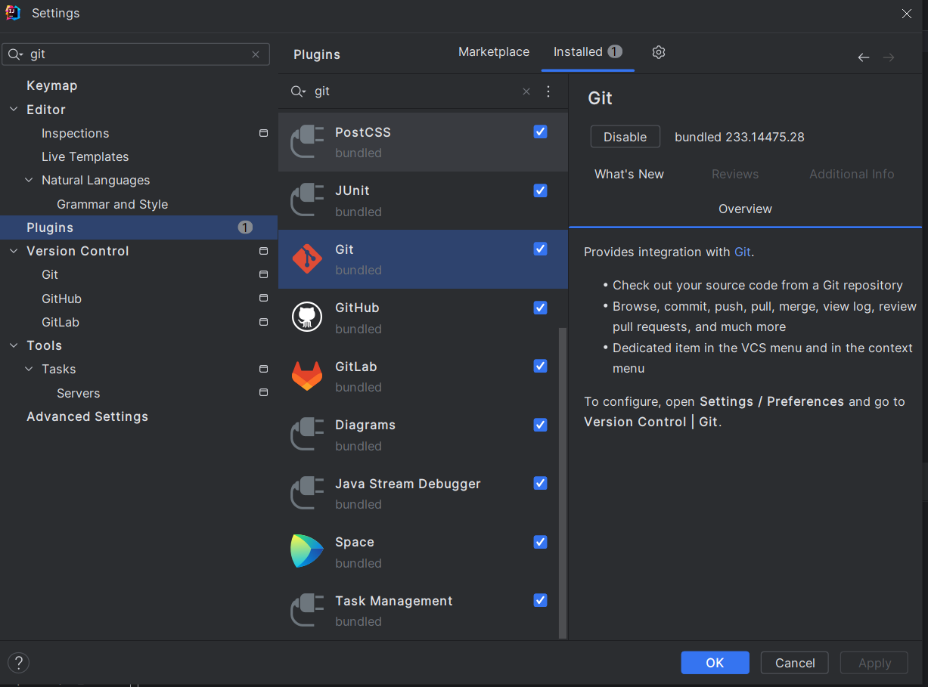


Рис.4 Контекстное меню Plugins

Шаг 2: Настройка Git в IntelliJ IDEA  
- После установки плагина Git откройте проект, с которым вы хотите работать  
- Выберите "VCS" в верхнем меню и затем "Enable Version Control Integration"  
- Выберите "Git" в списке систем контроля версий и нажмите "OK"

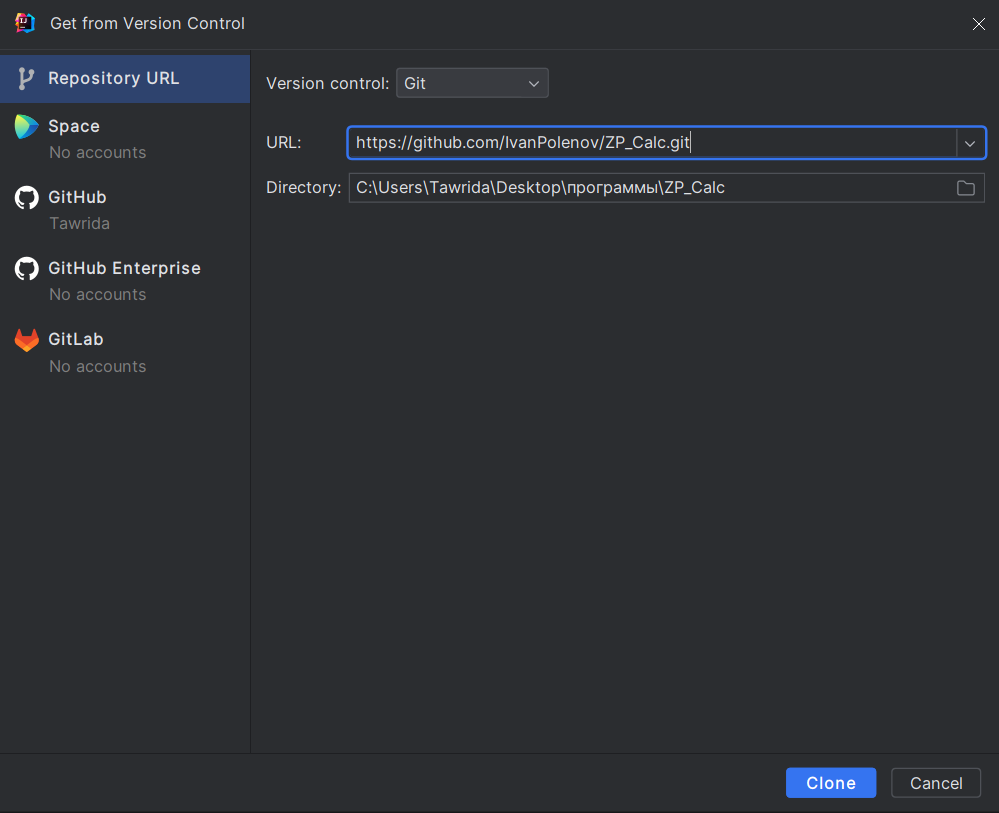


Рис. 5 Контекстное меню Еnable Version Control Integration

Шаг 3: Клонирование репозитория  
- Для клонирования репозитория выберите "VCS" -> "Git" -> "Clone"  
- Введите URL вашего репозитория Git и выберите путь для клонирования на вашем компьютере  
 Шаг 4: Работа с проектом в IntelliJ IDEA  
- После клонирования репозитория откройте проект в IntelliJ IDEA  
- Внесите необходимые изменения в файлы проекта  
- Добавьте изменения в индекс Git, выбрав файлы и используя контекстное меню или команду "Add to Git"  
- Сделайте коммит изменений, указав сообщение о коммите  
 Шаг 5: Отправка изменений на удаленный репозиторий  
- После коммита изменений выберите "VCS" -> "Git" -> "Push"  
- Введите учетные данные для доступа к вашему удаленному репозиторию  
 Теперь вы успешно подключились к Git и смогли внести изменения в проект через инструментальную среду IntelliJ IDEA.**5 Тестирование ПП**

При разработке ПО основной задачей является реализация заявленной функциональности и обеспечение качества ПО. Качество ПО – это совокупность характеристик ПО, относящихся к его способности удовлетворить установленные и предполагаемые потребности. Тестирование ПО – проверка соответствия между реальным и ожидаемым поведением программы, осуществляемая на конечном наборе тестов, выбранном определенным образом (SWEBOK).

Таблица с описанием UNIT-тестов и ответственных за их реализацию.

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **ФИО разработчика/модератора** | **Описание UNIT-теста** | **№ приложения** |
| 1 | Поленов И. С. | Проверка правильности ввода ввода. | см. Приложение П-1 |
| 2 | Тохтаров Д. И. | Проверка выбора формы. | см. Приложение П-2 |

Табл. 2 Описание UNIT-тестов

Заключение

В ходе выполнения курсовой работы была разработана программа “Зарплатного калькулятора. Повременная форма оплата труда” в IntelliJ IDEA были использованы современные инструменты разработки, такие как Git для контроля версий и управления кодом, а также функциональные возможности среды разработки IntelliJ IDEA для создания эффективного и удобного интерфейса.

Результатом проделанной работы является функциональная и интуитивно понятная программа, способная эффективно выполнять расчеты зарплаты на основе введенных данных. Разработка данного проекта позволила не только применить полученные знания и навыки, но и улучшить понимание принципов объектно-ориентированного программирования.

В целом, выполнение курсовой работы по разработке программы “Зарплатного калькулятора. Повременной формой оплаты труда” в IntelliJ IDEA было полезным опытом, позволившим поработать в команде через “социальную сеть”, используемую разработчиками GitHub.

## **ПРИЛОЖЕНИЕ П-1**

import java.util.InputMismatchException;  
import java.util.Scanner;  
public class SalaryCalculator {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.in);  
 // Проверка корректного ввода  
 try {  
 double hoursWorked = scanner.nextDouble();  
 double hourlyRate = scanner.nextDouble();  
 double[] result = calculateSalary(hoursWorked, hourlyRate);  
 System.out.println(STR."Ставка: \{result[0]}");  
 System.out.println(STR."НДФЛ: \{result[1]}");  
 System.out.println(STR."К выдаче: \{result[3]}");  
 } catch (InputMismatchException e) {  
 System.out.println("Ошибка.");  
 }  
 }  
// Функция для расчета зарплаты  
 public static double[] calculateSalary(double hoursWorked, double hourlyRate) {  
 if (hoursWorked < 0 || hourlyRate < 0) {  
 throw new IllegalArgumentException("Ошибка.");  
 }  
 double salary = hoursWorked \* hourlyRate;  
 double tax = 0.1 \* salary; // Налог 10%  
 double totalDeductions = tax;  
 double netSalary = salary - totalDeductions;  
 return new double[] {salary, tax, totalDeductions, netSalary};  
 }  
}

## **ПРИЛОЖЕНИЕ П-2**

import javax.swing.\*;  
import java.util.InputMismatchException;  
import java.util.Scanner;  
  
public class PaymentCalculator {  
  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.in);  
  
 // Проверка выбранной формы оплаты  
 boolean isDailyPayment = false;  
 AbstractButton radioButton1 = null;  
 AbstractButton radioButton2 = null;  
 if (radioButton1.isSelected()) {  
 isDailyPayment = true;  
 } else if (radioButton2.isSelected()) {  
 isDailyPayment = false;  
 } else {  
 System.out.println("Please select a payment type.");  
 return;  
 }  
  
 // Проверка корректного ввода  
 try {  
 double hoursWorked = scanner.nextDouble();  
 double hourlyRate = scanner.nextDouble();  
 double[] result = calculateSalary(hoursWorked, hourlyRate);  
 System.out.println(STR."Ставка: \{result[0]}");  
 System.out.println(STR."НДФЛ: \{result[1]}");  
 System.out.println(STR."К выдаче: \{result[3]}");  
 } catch (InputMismatchException e) {  
 System.out.println("Ошибка.");  
 }  
 }  
  
 }  
 // Функция для расчета зарплаты  
 public static double[] calculateSalary(double hoursWorked, double hourlyRate) {  
 if (hoursWorked < 0 || hourlyRate < 0) {  
 throw new IllegalArgumentException("Ошибка.");  
 }  
 double salary = hoursWorked \* hourlyRate;  
 double tax = 0.1 \* salary; // Налог 10%  
 double totalDeductions = tax;  
 double netSalary = salary - totalDeductions;  
 return new double[] {salary, tax, totalDeductions, netSalary};  
 }  
}

Список использованной литературы

1. Бердоносов, В. Д. Объектно-ориентированное программирование: учеб. пособие / В. Д. Бердоносов, А. А. Животова. – Комсомольск-на-Амуре: ФГБОУ ВПО «КнАГТУ», 2015. – 135 с.

2. Машнин, Т. С. Eclipse: разработка RCP-, Web-, Ajax- и Android-приложений на Java / Т. С. Машнин .— Санкт-Петербург : БХВ- Петербург, 2013 .— 384 с.

3. Коузен, К. Современный Java: рецепты программирования [Электронный ресурс] / Коузен К. — Москва : ДМК Пресс, 2018 .— 275 с. URL:https://e.lanbook.com/book/116121

4. Яшин, А. С. Java на примерах. Практика, практика и только практика [Электронный ресурс] / Яшин А. С., Сеттер Р. В. — Санкт-Петербург : Наука и Техника, 2018 .— 256 с. URL:https://e.lanbook.com/book/108278

5. Кулаков, К. А. Основы тестирования программного обеспечения [Электронный ресурс]: учебное электронное пособие для обучающихся Института математики и информационных технологий / К. А. Кулаков, В. М. Димитров; М-во образования и науки Рос. Федерации, Федер. гос. бюджет. образоват. учреждение высш. образования Петрозав. гос. ун-т. - Электрон. текст. дан. – Петрозаводск: Изд-во ПетрГУ, 2018.

6. The Eclipse Foundation, 2020. URL: https://www.eclipse.org/

7. Mantis Bug Tracker, 2020. URL: https://www.mantisbt.org/

8. Apache Subversion, 2020. URL: https://subversion.apache.org/

9. TortoiseSVN, 2020. URL: https://tortoisesvn.net/ (дата обращения: 02.10.2020)

10. Maven – Welcome to Apache Maven, 2020. URL: https://maven.apache.org/