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-233бз

Гареев В.С.

Фарвазетдинова

К.Р.

Проверил: преподаватель Казанцев А.В.

г. Уфа 2024

**Министерство науки и 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 | *Гареев В.С*. | Группа | *ПИ-233Бз* | Консультант | *Казанцев А.В.* |
|  | Фамилия И.О. |  | номер группы |  | Фамилия И.О. |
| Студент | *Фарвазетдинова К.Р* | Группа | *ПИ-233Бз* | Консультант | *Казанцев А.В.* |
|  | Фамилия И.О. |  | номер группы |  | Фамилия И.О. |

|  |  |
| --- | --- |
| 1.Тема КР: | *Клининговый калькулятор услуг. Офис.* |
|  | наименование темы |
| 2.Основное содержание: | |
| 1. Пояснительная записка с необходимыми материалами. 2. Репозиторий системы контроля версий содержащий программный код с комментариями. | |

1. Требования к оформлению:
   1. Пояснительная записка должна быть оформлена в текстовом процессоре LibreOffice Writer в соответствии с требованиями ГОСТ. В бумажном виде оформляются: титульный лист, бланк задания, также прикладывается аннотация, где указывается ссылка на репозиторий, содержащий исходный проект, работающую программу и пояснительную записку.
2. Графическая часть должна включать:

* экранные формы инструментальных средств;
* экранные формы, разрабатываемого программного продукта.

Дата выдачи 28.10.2023 Дата окончания 27.01.2024

Руководитель *Казанцев А.В.* Студент *Гареев В.С*.

Студент *Фарвазетдинова К.Р*

СОДЕРЖАНИЕ

[1. Описание предметной области 5](#_Toc157217007)

[2. Описание прототипа ПП 7](#_Toc157217008)

[3. Математическая модель работы ПП 9](#_Toc157217009)

[4. Написание программного кода по зонам ответственности 11](#_Toc157217010)

[5. Тестирование ПП 14](#_Toc157217011)

[Заключение 15](#_Toc157217012)

[Приложение П-1 16](#_Toc157217013)

[Приложение П-2 17](#_Toc157217014)

[Список использованной литературы 18](#_Toc157217015)

# Описание предметной области

Необходимо разработать систему для расчета стоимости клининговых услуг для офисных помещений. В данной системе должны быть рассмотрены и учтены основные аспекты, влияющие на расчет стоимости клининга в офисном помещении, такие как:

* площадь помещения;
* тариф уборки;
* дополнительные услуги (уборка полов, мебели, окон и т.д.);

Также следует рассмотреть вопросы автоматизации процесса расчета стоимости клининга, разработки удобного интерфейса для пользователей и другие аспекты, влияющие на эффективность использования калькулятора клининговых услуг в офисном помещении.

Для успешной разработки системы для расчета стоимости клининговых услуг для офисных помещений необходимо уделить внимание нескольким ключевым аспектам. Во-первых, следует провести анализ рынка клининговых услуг, изучить спрос и предложение на данном сегменте, выявить основные конкурентные преимущества и недостатки существующих решений.

Затем необходимо изучить основные факторы, влияющие на расчет стоимости клининга в офисном помещении. Это включает в себя не только площадь помещения и тариф уборки, но и дополнительные услуги, которые могут быть востребованы заказчиками, такие как уборка полов, мебели, окон и другие поверхностей.

Кроме того, важным аспектом является разработка удобного интерфейса для пользователей, который позволит заказчикам быстро и эффективно рассчитывать стоимость клининговых услуг для своих офисных помещений. Автоматизация процесса расчета также может значительно повысить эффективность использования калькулятора клининговых услуг.

Для оптимизации процесса расчета стоимости клининга для офисов необходимо учитывать специфику офисных помещений, их типовые особенности и требования к уровню чистоты. Также важно учесть потребности заказчиков и предложить гибкие условия расчета, которые позволят удовлетворить разнообразные запросы.

Наконец, разработка системы для расчета стоимости клининговых услуг для офисных помещений должна быть ориентирована на создание надежного и точного инструмента, который будет способствовать упрощению процесса заказа клининговых услуг и повышению удовлетворенности клиентов.

# Описание прототипа ПП

Калькулятор клининговых услуг для офисных помещений — это программа, которая может быть использована заинтересованным лицом для расчёта стоимости клининговых услуг в офисном помещении. Данный калькулятор должен иметь удобный и интуитивно-понятный интерфейс пользователя, который бы включал в себя следующие возможности:

- выбор тарифа уборки;

- ввод площади офисного помещения;

- выбор дополнительных услуг (уборка полов, мебели, окон и т.д)

Итоговая стоимость услуг должна выводится в понятном пользователю формату чтобы не вводить его в заблуждение. Главная экранная форма калькулятора клининговых услуг, показана на рисунке 1.

Приведем описание основных компонентов, используемых на главной

экранной форме:

1. Все надписи (пояснения) являются метками с надписями (JLabel).
2. Компонент является выпадающим списком с возможностью выбора одного из вариантов уборки из списка (JComboBox). Для выбора доступны три варианта тарифов уборки: Эконом, Бизнес и Vip.
3. Компонент является тестовым полем с возможностью редактирования (JTextField). В данное поле вводится площадь офисного помещения
4. Компонент является переключателями (JCheckBox), которые позволяют указать какие пользователю необходимы дополнительные услуги. Доступны варианты: Уборка ванн и туалетов, Чистка ковров, Мойка окон, Дезинфекция, Уборка после ремонта, Чистка мебели, Уборка кухни, Уборка вентиляционных систем
5. Компонент является кнопкой (JButton) с реализацией обработчика событий – расчет стоимости клининговых услуг

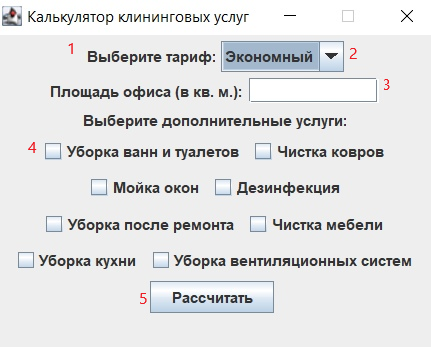


Рис. 1. Главная экранная форма

# Математическая модель работы ПП

Математическая модель работы программы «Калькулятор клининговых услуг в офисном помещении».

Стоимость клининговых услуг рассчитывается по формуле 1.

– это итоговая сумма клининговых услуг

N – стоимость уборки по тарифу за 1 кв/м

S – площадь офисного помещения

- сумма стоимости дополнительных услуг

Рассмотрим несколько примеров расчета. В первом случае будут выбраны следующие значения:

N = 135.50 рублей за 1 кв/м (соответствует тарифу «Эконом»)

S = 105 кв/м

Дополнительные услуги:

Уборка ванной и туалетов = 500 рублей

Мойка окон = 750 рублей

Тогда итоговая сумма будет равна:

= 15477.5 рублей

Во втором:

N = 352.52 рублей за 1 кв/м (соответствует тарифу «Бизнес»)

S = 251 кв/м

Дополнительные услуги:

Уборка ванной и туалетов = 500 рублей

Чистка ковров = 600 рублей

Тогда итоговая сумма будет равна:

= 89582.52 рубля

И в третьем случае:

N = 1000 рублей за 1 кв/м (соответствует тарифу «Бизнес»)

S = 140 кв/м

Дополнительные услуги:

Мойка окон = 750 рублей

Тогда итоговая сумма будет равна:

= 140750 рублей

# 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 Eclipse IDE.

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

Таблица 1 – Зоны ответственности

|  |  |  |
| --- | --- | --- |
| **№** | **ФИО разработчика/модератора** | **Зона ответственности** |
| 1 | Гареев Вадим Сергеевич | Разработка и реализация интерфейса главного окна программы (Main.java).  Разработка функционала главного окна программы (Main.java) |
| 2 | Фарвазетдинова Карина Радиковна | Разработка абстрактного класса Product.java с полями (price, name) и абстрактным методами toString() и getPrice(). Реализация абстрактного класса через классы AdditionalServices.java, Rate.java  Создание класса Application.java которые содержит в себе данные необходимы для расчета и метод Calculate() который производит расчет на основании введенных данных |

Для того чтобы подключиться к проекту используя систему контролей версий git необходимо следующее:

1. Зарегистрироваться на сайте GitHub и направить модератору свои данные чтобы он мог выдать вам доступ к репозиторию
2. Далее необходимо клонировать репозиторий на свой компьютер, для этого щелкните правой кнопкой мыши по панели «Git Repositories» в Eclipse IDE и выберите пункт «Clone а Git Repository…» как показано на рисунке 2.



Рис. 2. Клонирование репозитория

1. После чего выберите пункт «Clone URL» в появившемся окне которое показано на рисунке 3. Далее нажмите на кнопку «Next»

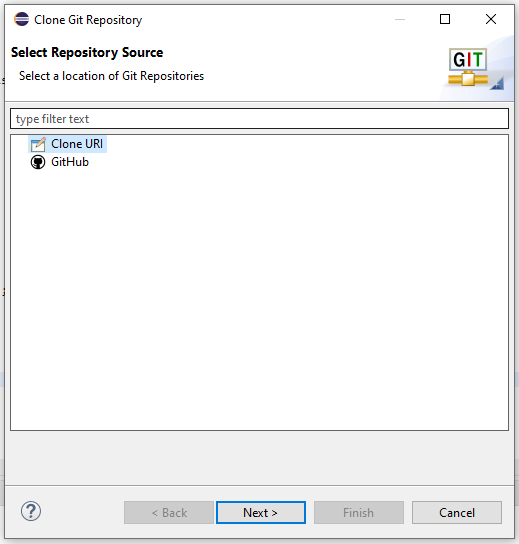


Рис. 3. Выбор источника репозитория

1. После чего заполните в следующем окне (рисунок 4) все необходимые данные и нажмите на кнопку «Finish»

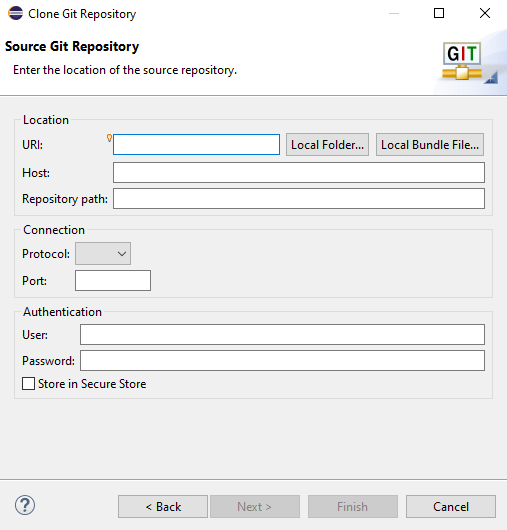


Рис. 4. Форма заполнения данных для получения репозитория

# Тестирование ПП

При разработке программного обеспечения основной целью является реализация заявленной функциональности и обеспечение высокого качества продукта.

Качество программного обеспечения - это совокупность характеристик, относящихся к способности ПО удовлетворить установленные и предполагаемые потребности.

Тестирование программного обеспечения - это процесс проверки соответствия между реальным и ожидаемым поведением программы, осуществляемый на конечном наборе тестов, выбранных определенным образом.

JUnit - это фреймворк для тестирования программного обеспечения на языке Java. С его помощью разработчики могут создавать и запускать автоматизированные тесты, которые проверяют правильность работы отдельных частей программы.

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **ФИО разработчика/ модератора** | **Описание JUnit -теста** | **№ приложения** |
| 1 | Гареев Вадим Сергеевич | Проверяет работу главного окна приложения | см. Приложение П-1 |
| 2 | Фарвазетдинова Карина Радиковна | Проверяет работу метода по подсчету суммы клининговых услуг | см. Приложение П-2 |

# Заключение

В ходе выполнения разработки проекта «Калькулятор клининговых услуг. Офис» были проведены исследования, анализ требований пользователей и разработано эффективное приложение, способное упростить процесс расчета стоимости клининговых услуг. Разработанное приложение предоставляет пользователям удобный интерфейс, позволяющий вводить различные параметры и получать точные расчеты стоимости услуг.

Пользователи могут легко настраивать приложение под свои потребности, выбирая необходимые параметры и получая мгновенные расчеты. В результате, разработанное ПО значительно упростило процесс подсчета стоимости на клининговые услуги в офисе, что сделало его востребованным среди компаний и предприятий.

Благодаря использованию современных технологий и оптимизированных алгоритмов, проект «Калькулятор клининговых услуг. Офис» обеспечивает точные расчеты и позволяет пользователям быстро настраивать параметры в соответствии с их потребностями. В результате, данное ПО становится важным инструментом для компаний и предприятий, помогая им эффективно управлять ресурсами и бюджетированием клининговых услуг.

# **Приложение П-1**

package Core;

import static org.junit.Assert.assertEquals;

import javax.swing.\*;

import Classes.Product;

import org.junit.Before;

import org.junit.Test;

public class MainTest {

private JPanel myPanel;

@Before

public void setUp() {

myPanel = new Main().mainPanel;

}

@Test

public void testComboBoxAndCheckBox() {

JComboBox<Product> comboBox = (JComboBox<Product>) myPanel.getComponent(1);

comboBox.setSelectedIndex(1);

JTextField textField = (JTextField) myPanel.getComponent(3);

textField.setText("50");

JCheckBox checkBox = (JCheckBox) myPanel.getComponent(6);

checkBox.setSelected(true);

JButton button = (JButton) myPanel.getComponent(13);

button.doClick();

JLabel label = (JLabel) myPanel.getComponent(14);

assertEquals("Стоимость уборки составит: 18226,00 рублей", label.getText());

}

}

# **Приложение П-2**

package Classes;

import static org.junit.Assert.\*;

import java.util.ArrayList;

import java.util.List;

import org.junit.Test;

public class ApplicationTest {

@Test

public void test() {

Product eco\_tariff = new Rate(135.50, "Экономный");

Product services1 = new AdditionalServices(500, "Уборка ванн и туалетов");

Product services2 = new AdditionalServices(600, "Чистка ковров");

List<Product> services = new ArrayList<>();

services.add(services1);

services.add(services2);

Application newApplic = new Application(50, eco\_tariff, services);

double result = newApplic.Calculate();

assertTrue("Неверно подсчитано значение суммы", result == 7875);

}

}

# 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/