Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский государственный технический университет имени Н.Э. Баумана»

(национальный исследовательский университет)

Московский техникум космического приборостроения

09.02.07 Информационные системы и программирование

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №9**

по МДК 02.01. Технология разработки программного обеспечения

Построение архитектуры программного средства

|  |  |
| --- | --- |
| Выполнил студент | П.Р. Симонян |
| Курс \_\_\_\_\_\_3\_\_\_\_\_\_Группа\_\_\_\_\_ТИП-51\_\_\_\_ |  |
| Подпись студента\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Е.А. Митрошенкова |

Москва 2021

Оглавление

[1. Тема 3](#_Toc84801124)

[2. Постановка задачи 4](#_Toc84801125)

[3. Диаграмма последовательности 5](#_Toc84801126)

[4. Диаграмма кооперации 6](#_Toc84801127)

[5. Ответы на контрольные вопросы 7](#_Toc84801128)

# 1. Тема

**Лабораторная работа №9**

**Тема**: Создание диаграммы классов для сценария «Добавить и оплатить новый заказ» прецедента «Работа с заказом».

**Вариант:** страховая компания

**Цель**: получить навыки построения диаграммы компонентов

**Оборудование**: персональный компьютер

**Программное обеспечение**: Microsoft Word, Visio

# 2. Постановка задачи

1. Ознакомиться с теоретическими сведениями.
2. Ознакомиться с методическими рекомендациями и выполнить задание в соответствии со своим вариантом.
3. Построить диаграмму классов
4. Для каждого класса необходимо задать атрибуты и операции. Каждый класс должен быть подробно задокументирован - необходимо задать текстовое описание самого класса (см. пример) имя самого класса, описания его атрибутов, типов их значений и операций. Также должны быть определены кванторы видимости для атрибутов и операций.
5. Ответить на контрольные вопросы
6. Написать отчет

# 3. Ход работы

1. Цель работы – создать диаграмму классов для указанного ранее случая.

2. Диаграмма классов будет рассматриваться с точки зрения разработки программного обеспечения.

3. Создадим диаграмму классов, выделив следующие сущности:

1) Пользователь (в том числе унаследованные сущности клиента и агента)

2) Заказ

3) Услуга

4. Ниже представлено описание классов на диаграмме:

**Класс User:**

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс, представляющий собой любого зарегистрированного пользователя системы, а также набор операций для работы с ними и создания новых пользователей. |
| Атрибуты | +login: String – псевдоним пользователя в системе;  #password: String – пароль пользователя  +privilege: String – тип пользователя и смежные с ним привилегии |
| Операции | +CreateUser(): User – операция для создания нового пользователя  +Login(): User – операция для входа в систему  +ChangePassword(): User – операция для изменения пароля |

**Класс Client:**

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс, описывающий клиента, доступные только ему операции и, в случае расширения, уникальные для него атрибуты |
| Атрибуты | Унаследованные от класса User |
| Операции | Унаследованные от класса User +  +MakeOrder(): Order – операция для создания заказа и связанного с ним договора |

**Класс Agent:**

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс, описывающий страхового агента и имеющий доступные только для него операции и атрибуты (в случае расширения системы) |
| Атрибуты | Унаследованные от класса User |
| Операции | Унаследованные от класса User +  +SendReport (): bool – отправить отчет о деятельности агента |

**Класс Order:**

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс, описывающий договор и связанный с ним заказ |
| Атрибуты | +id: int – номер договора  +id\_Client: int – номер клиента  +id\_Agent: int – номер агента  +orderState: string = ‘Договор не подписан’ – состояние исполнения договора |
| Операции | +SignOrder() – операция подписания договора  +Pay() – операция оплаты заказа |

**Класс Service:**

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс, описывающий услугу страхования |
| Атрибуты | +id: int – номер услуги  +description: string – описание услуги  +price: float – стоимость услуги |
| Операции |  |

5. Добавим отношения между классами:

User : Client/Agent – отношение обобщения.

Client : Order/Agent – зависимость, поскольку заказ зависит от классов Client и Agent

Order : Service – композиция, поскольку Order включает в себя множество услуг (связь 1 ко многим), но услуги могут существовать в системе сами по себе.

6. Диаграмма классов, созданная по описанию выше, представлена на рисунке 3.1.



Рисунок 3.1 – Диаграмма классов

# 4. Ответы на контрольные вопросы

**1. Назначение и понятие диаграммы классов?**

Диаграмма классов позволяет описать статическую структуру моделируемой программной системы. Она описывает множество групп связанных объектов, объединенных в классы.

**2. Опишите категории связей между классами.**

Агрегация – зависимые один класс является частью другого, но может существовать и сам по себе.

Композиция – один класс является частью другого, но они не могут существовать друг без друга.

Обобщение – один класс наследуется от другого.

Зависимость – один класс меняется при изменении другого.

**3. Понятие класса?**

Класс – описание множества объектов с общими атрибутами, определяющими состояние, и операциями, определяющими поведение.

**4. Что такое атрибуты класса?**

Атрибуты класса – свойства или переменные, которые описывают состояние класса или его экземпляров. Они бывают различных типов, например String (строка) или float (число с плавающей запятой).

**5. Что означает видимость элементов класса и как она обозначается на диаграмме?**

Видимость элементов класса определяет, кто может использовать этот элемент.

+ перед элементом класса – элемент доступен всем

# – элемент доступен классу и его потомкам

‘-‘ – элемент доступен только этому классу и его экземплярам.