Министерство науки и высшего образования Российской Федерации

федеральное государственное бюджетное образовательное учреждение высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

МОРДОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

им. Н.П. ОГАРЁВА»

(ФГБОУ ВО «МГУ им. Н.П. Огарёва»)

Факультет математики и информационных технологий

Кафедра систем автоматизированного проектирования

ОТЧЁТ О ЛАБОРАТОРНОЙ РАБОТЕ №4

по дисциплине: Программная инженерия

МЕТОДОЛОГИЯ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ

Автор отчёта о лабораторной работе А. Е. Конышев подпись, дата

Обозначение лабораторной работы ЛР–02069964–02.03.02–08–24

Направление подготовки 02.03.02 Фундаментальная информатика и информационные технологии

Руководитель работы

преподаватель кафедры

систем автоматизированного

проектирования Ю. С. Кочкаева подпись, дата

Саранск 2024

# Лабораторная работа № 4

# «Методология объектно-ориентированного моделирования»

# Цель работы:

Ознакомление с основными элементами определения, представления, проектирования и моделирования программных систем с помощью языка UML.

# Порядок выполнения работы

1. Изучить предлагаемый теоретический материал.
2. Постройте диаграмму вариантов использования для выбранной информационной системы.
3. Выполните реализацию вариантов использования в терминах взаимодействующих объектов и представляющую собой набор диаграмм:

* диаграмм классов, реализующих вариант использования;
* диаграмм взаимодействия (диаграмм последовательности и кооперативных диаграмм), отражающих взаимодействие объектов в процессе реализации варианта использования.

1. Разделить классы по пакетам использую один из механизм разбиения.
2. Постройте диаграмму состояний для конкретных объектов информационной системы.
3. Построить отчёт, включающий все полученные уровни модели, описание функциональных блоков, потоков данных, хранилищ и внешних объектов.

# Выполнение

# Диаграмма вариантов использования.

Диаграмма вариантов использования для автоматизированной информационной системы «Туристическое агентство» приведена на рисунке 1.

Для данной АИС выделяется 2 действующих лица: пользователь-клиент и пользователь-администратор. Так же требованиями заказчика определено несколько видов деятельности акторов: клиенту должны предоставляться возможности регистрации, входа в систему, осуществления заказа; админнистратор должен иметь возможность входа в систему, создания/редактирования/удаления туров, при условии авторизации в системе.

На диаграмме использованы отношения обобщения, указывающие на наличие общих черт у действующих лиц, отношения ассоциации, отражающие возможность использования актором прецедента и отношения включения, указывающее на то, что что включаемый прецедент должен быть обязательным для дополняемого.

Пользователь

Клиент

Администратор

Сделать заказ

Зарегистрироваться

Войти в систему

Создать туры

Редактировать туры

Удалить туры

<<include>>

<<include>>

Открыть доступные туры

<<include>>

Рисунок 1 – Диаграмма вариантов использования

# Диаграммы взаимодействия

# Диаграмма последовательности

Диаграмма последовательности отражает поток событий, происходящих в рамках варианта использования. Она изображена на рисунке 2.

Все действующие лица показаны в верхней части диаграммы:

1. клиент туристического агентства,
2. окно входа,
3. кнопка входа,
4. окно навигации,
5. кнопка, открывающая форму туров,
6. форма туров,
7. кнопка, открывающая конкретный тур,
8. форма тура,
9. кнопка заказа,
10. форма заказа,
11. кнопка оформления заказа.

Стрелки соответствуют сообщениям, передаваемым между действующим лицом и объектом или между объектами для выполнения требуемых функций.

На диаграмме последовательности объекты изображены ввиде прямоугольников, от которых вниз проведена вертикальная линия. Эта линия называется линией жизни (lifeline) объекта. Она представляет собой фрагмент жизненного цикла объекта в процессе взаимодействия.

# Каждое сообщение представлено в виде стрелки между линиями жизни двух объектов. Сообщения появляются в том порядке, как они показаны на странице сверху вниз. Каждое сообщение помечается номером и именем сообщения:

# Ввод логина и пароля,

# Передача логина и пароля в код VBA,

# Проверка корректности введенных логина и пароля,

# Отрытие пользовательского окна навигации,

# Запрос действия,

# Выбор действия (открыть окно всех туров),

# Ооткрыть окно всех туров,

# Запрос действия,

# Выбор действия (открыть конкретный тур),

# Открыть конкретный тур,

# Запрос действия,

# Выбор действия (заказать тур),

# Открыть окно заказа,

# Запросить информацию, необходимую для оформления заказа,

# Ввод информации, необходимой для оформления заказа,

# Передача введенной информации коду VBA,

# Проверка корректности введенной информации,

# Сообщение об ошибке или об успешном оформлении заказа.

# 

Рисунок 2 – Диаграмма последовательности

Клиент

Форма  
Входа

Кнопка входа

Окно навигации

Кнопка туров

Форма туров

Кнопка тура

Форма тура

Кнопка заказа

Форма заказа

Кнопка оформления заказа

1

2

3

4

5

6

8

9

7

10

11

12

13

14

15

16

17

18

# Диаграмма кооперации

Диаграмма кооперации изображена на рисунке 3. Она показывает связи между объектами, перечисленными в пункте 2.1 (диаграммы последовательности).

На диаграмме кооперации представлена вся та информация, которая есть и на диаграмме последовательности, но кооперативная диаграмма по-другому описывает поток событий. Из нее легче понять связи между объектами, однако, труднее уяснить последовательность событий.

На кооперативной диаграмме так же, как и на диаграмме последовательности, стрелки обозначают сообщения, обмен которыми осуществляется в рамках данного варианта использования. Их временная последовательность указывается путем нумерации сообщений.

# 

Форма заказа

Кнопка оформления заказа

Форма  
Входа

Кнопка входа

Окно навигации

Кнопка туров

Форма туров

Кнопка тура

Форма тура

Кнопка заказа

Форма заказа

Кнопка оформления заказа

Клиент

Рисунок 3 – Кооперативная диаграмма

# Диаграммы классов

Диаграмма классов определяет типы классов системы и различного рода статические связи, которые существуют между ними. На диаграммах классов изображаются также атрибуты классов, операции классов и ограничения, которые накладываются на связи между классами.

*Диаграмма классов UML* - это граф, узлами которого являются элементы статической структуры проекта (классы, интерфейсы), а дугами - отношения между узлами (ассоциации, наследование, зависимости).

На диаграмме классов изображаются следующие элементы:

* *пакет* (*package*) - набор элементов модели, логически связанных между собой;
* *класс* (*class*) - описание общих свойств группы сходных объектов;
* *интерфейс* (*interface*) - абстрактный класс, задающий набор операций, которые объект произвольного класса, связанного с данным интерфейсом, предоставляет другим объектам.

*Класс* – это группа сущностей (объектов), обладающих сходными свойствами, а именно, данными и поведением. Отдельный представитель некоторого класса называется объектом класса или просто объектом.

Под поведением объекта в UML понимаются любые правила взаимодействия объекта с внешним миром и с данными самого объекта.

На диаграммах класс изображается в виде прямоугольника со сплошной границей, разделенного горизонтальными линиями на 3 секции:

* *верхняя секция* (*секция имени*) содержит имя класса и другие общие свойства (в частности, стереотип);
* в средней секции содержится список атрибутов;
* в нижней – список операций класса, отражающих его поведение (действия, выполняемые классом).

Классы разделены по функциональности на 4 пакета (см. рисунок 4 – рисунок 8):

* вход в систему,
* навигация,
* каталог туров,
* тур,
* заказы.

Между классами используются связи агрегации, ассоциации, наследования и зависимости.

На рисунке 9 изображена диаграмма пакетов.

Entity

Пользователи

Id

ФИО

Логин

Пароль

Адрес

Возраст

Пол

Гражданство

Конструктор

Деструктор

Boundary

LoginForm

SetLogin

GetLogin

SetPassword

GetPassword

Control

ControlLogin

CheckLogin

CheckPassword

OpenForm

ErrorMessages

Cancel

Рисунок 4 – Диаграмма классов. Пакет Вход в систему

ChoiceID

Boundary

ToursForm

GethoiceID

Control

ControlToursForm

Set ChoiceID

Get ChoiceID

ErrorMessages

OpenTourForm

Cancel

Рисунок 5 – Диаграмма классов. Пакет Каталог туров

Id

Наименование

Страна

Транспорт

Пакет услуг

Вариант размещения

Стоимость

Entity

Туры

Конструктор

Деструктор

Тур

Boundary

ToursForm

PrintTour

Control

ControlToursForm

Set Tour

Get Tour

CheckTours

ErrorMessages

OpenOrerForm

Cancel

Рисунок 6 – Диаграмма классов. Пакет Тур

Рисунок 7 – Диаграмма классов. Пакет Навигация

Id

Наименование

Страна

Транспорт

Пакет услуг

Вариант размещения

Стоимость

Entity

Туры

Конструктор

Деструктор

ChoiceID

Boundary

NavigationForm

GetТуры  
GetПакетыУслуг  
GetТранспорт  
GetРазмещение  
GetСтраны

Control

ControlNavigation

Set ChoiceID

Get ChoiceID

ErrorMessages

OpenToursForm

Cancel

Entity

Транспорт

Id

Модель

Тип транспорта

Транспорт

Собственник

Стоимость

Конструктор

Деструктор

Entity

Размещение

Id

Наименование

Тип размещения

Транспорт

Собственник

Стоимость

Оценка

Конструктор

Деструктор

Id

Наименование

Описание

Стоимость

Entity

Пакеты услуг

Конструктор

Деструктор

Id

Тип транспорта

Entity

Типы транспорта

Конструктор

Деструктор

Entity

Типы размещения

Id

Тип размещения

Конструктор

Деструктор

Entity

Услуги

Id

Наименование

Описание

Стоимость

Конструктор

Деструктор

Entity

Типы размещения

Id

Тип размещения

Конструктор

Деструктор

Entity

Пользователи

Id

ФИО

Логин

Пароль

Адрес

Возраст

Пол

Гражданство

Конструктор

Деструктор

Boundary

OrdersForm

SetCustomer

GetCustomer

SetTour

GetTour

Control

Control OrdersForm

SetStatusOrder  
GetStatusOrder

CheckCustomer

CheckTour

ErrorMessages

SuccessMessage

Cancel  
WriteOrder

Id

Наименование

Страна

Транспорт

Пакет услуг

Вариант размещения

Стоимость

Entity

Туры

Конструктор

Деструктор

Entity

Заказы

Id

Пользователь

Тур

Статус Заказа

Конструктор

Деструктор

Рисунок 8 – Диаграмма классов. Пакет Заказы

Вход в систему

Навигация

Каталог туров

Туры

Заказы

Рисунок 9 – Диаграмма пакетов

# Диаграмма состояний

Диаграммы  состояний  определяют  все  возможные  состояния, в которых может находиться конкретный объект, а также процесс смены состояний  объекта  в  результате  наступления  некоторых  событий.

Диаграмма состояний приведена на рисунке 10.

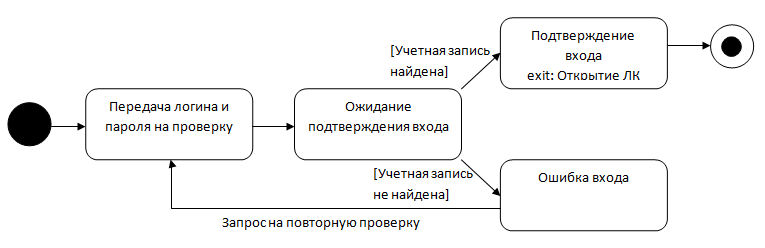


Рисунок 10 – Диаграмма состояний