«Белорусский государственный технологический университет»

«Кафедра программной инженерии»

Лабораторная работа № 2

По дисциплине: «Проектирование интернет-систем»

Тема: «Объектно-ориентированное моделирование в UML.  
Диаграммы прецендентов.»

Выполнил: Иконов Василий

4 к, 4гр.

Минск, 2019 г.

**1. Цель работы:**

*Изучить методологии объектно-ориентированного моделирования в UML.* Лабораторная работа направлена на ознакомление с основными принципами разработки программного обеспечения, выполнение базовых шагов проектирования  
*функциональности* информационной системы с применением методологии UML.

**2. Ответы на контрольные вопросы:**

1) Перечислите основные диаграммы языка *UML 2.0.*

* диаграмма классов;

**Диаграмма классов** — структурная диаграмма языка моделирования UML, демонстрирующая общую структуру иерархии классов системы, их коопераций, атрибутов (полей), методов, интерфейсов и взаимосвязей между ними.

* диаграмма поведения;
* схема последовательностей

Диаграммы последовательности (sequence diagram) являются видом диаграмм взаимодействия языка UML, которые описывают отношения объектов в различных условиях. Условия взаимодействия задаются сценарием, полученным на этапе разработки диаграмм вариантов использования.

* нотация базы данных;

Описание объектов предметной области, представляемых в логической модели базы данных в виде сущностей, в нотации и выполняется с помощью классов , где наименование класса представляется в верхней (первой) части обозначения, а атрибутивный состав размещается во втором блоке обозначения класса.

**Диагра́мма де́ятельности**— UML-диаграмма, на которой показаны действия, состояния которых описано на диаграмме состояний. Под деятельностью понимается спецификация исполняемого поведения в виде координированного последовательного и параллельного выполнения подчинённых элементов — вложенных видов деятельности и отдельных действий, соединённых между собой потоками, которые идут от выходов одного узла ко входам другого. Диаграммы деятельности используются при моделировании бизнес-процессов, технологических процессов, последовательных и параллельных вычислений.

* Диаграмма состояний   
  Диаграмма состояний (Statechart diagram) показывает автомат, фокусируя внимание на потоке управления от состояния к состоянию. Автомат (State machine) -это описание последовательности состояний, через которые проходит объект на протяжении своего жизненного цикла, реагируя на события, - в том числе описание реакций на эти события. Состояние (State) - это ситуация в жизни объекта, на протяжении которой он удовлетворяет некоторому условию, осуществляет определенную деятельность или ожидает какого-то события. Событие (Event) - это спецификация существенного факта, который происходит во времени и пространстве. В контексте автоматов событие - это стимул, способный вызвать срабатывание перехода. Переход (Transition) - это отношение между двумя состояниями, показывающее, что объект, находящийся в первом состоянии, должен выполнить некоторые действия и перейти во второе состояние, как только произойдет определенное событие и будут выполнены заданные условия. Деятельность (Activity) -это продолжающееся неатомарное вычисление внутри автомата. Действие (Action) - это атомарное вычисление, которое приводит к смене состояния или возврату значения. Диаграмма состояний изображается в виде графа с вершинами и ребрами.

2) Перечислите известные CASE-средства создания UML диаграмм.

**CASE, Computer-Aided Software Engineering**- набор инструментов и методов программной инженерии для проектирования программного обеспечения, который помогает обеспечить высокое качество программ, отсутствие ошибок и простоту в обслуживании программных продуктов.

* IBM Rational Rose
* Borland Together
* Microsoft Visio
* Sparx Systems Enterprise Architect
* Gentleware Poseidon
* SmartDraw
* Dia
* Telelogic TAU G2
* StarUML

3) Укажите назначение диаграммы вариантов использования.

Назначение данной диаграммы состоит в следующем: проектируемая программная система представляется в форме так называемых *вариантов использования*, с которыми взаимодействуют *внешние сущности* или *актеры*. При этом *актером* или действующим лицом называется любой *объект*, субъект или система, взаимодействующая с моделируемой бизнес-системой извне. Это может быть человек, техническое устройство, *программа* или любая другая система, которая служит источником воздействия на моделируемую систему так, как определит разработчик. *Вариант использования* служит для описания сервисов, которые система предоставляет *актеру*. Другими словами каждый *вариант использования* определяет набор действий, совершаемый системой при диалоге с *актером*. При этом ничего не говорится о том, каким образом будет реализовано взаимодействие *актеров* с системой и собственно выполнение *вариантов использования*.

4) Дайте описание нотаций, которые используются для построения диаграммы прецедентов.

**Актер** – это типовые пользователи (менеджер, операторы и т.д). Актер представляет собой некоторую роль, которую играет пользователь (или другая система) по отношению к системе.

**Отношение включения** - используется, когда имеется какой-либо фрагмент поведения системы, которые повторяется более чем в одном варианте использования и необходимо, чтоб его описание копировалось в каждом из этих вариантов использования.

**Отношение обобщения** – если есть один вариант использования, подобный другому, но намного шире (более глобальный).

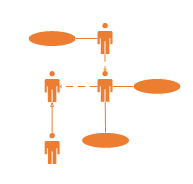
**Отношение расширения** – расширяющий вариант использования может дополнят (делать более точным) поведение базового вариант использования, определив в базовом варианте использования точки расширения.

**3. Постановка задачи:**

|  |  |
| --- | --- |
| Система аутентификации | Проектируемая **ИС** предназначена для проверки прав доступа пользователя. Система позволяет пользователю вести «ключ» (данные для аутентификации), затем выполняет поверку права доступа, после чего выводит результат проверки пользователю. Работа системы выполняется без администратора по ранее заданным параметрам аутентификации. Реализовать один интерфейс и БД |

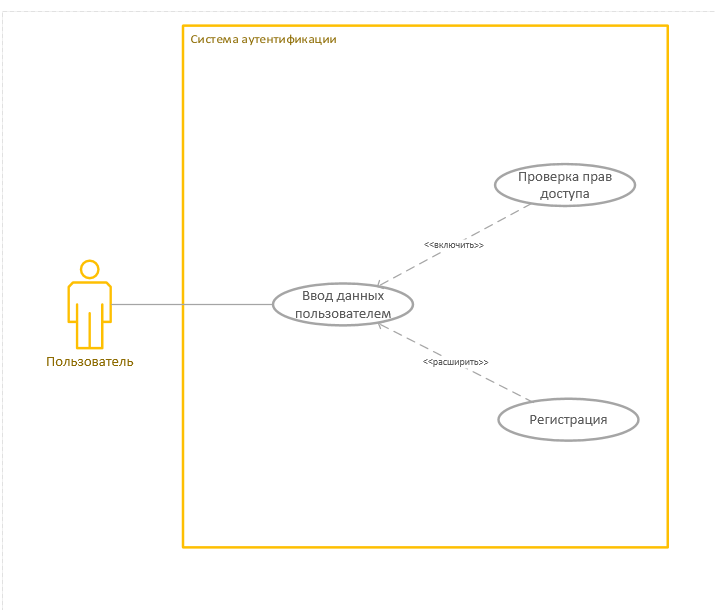
**4. Описание программно-аппаратных средств, используемые при выполнении работы:**

UML – диаграмма создана в среде разработки Visio-2016.



**5. Основная часть:**

**USE-CASE диаграмма**



**6. Выводы.**

В данной лабораторной работе я ознакомился с основными принципами разработки программного обеспечения, выполнение базовых шагов проектирования функциональностиинформационной системы с применением методологии UML. Мною была построена use-case диаграмма поведения для системы аутентификации пользователей.