Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Институт информационных технологий

Объектно-ориентированные технологии программирования и стандарты проектирования

«Ювелирная мастерская»

Выполнил: студент гр. 581061 Фут Д.С.

Проверила: Бакунова О.М.

Минск 2017

**Постановка задачи**

Ювелирная мастерская осуществляет изготовление ювелирных изделий для частных лиц на заказ. Мастерская работает с определенными материалами (платина, золото, серебро, различные драгоценные камни и т.д.). При обращении потенциального клиента в мастерской определяются с тем, какое именно изделие ему необходимо. Все изготавливаемые изделия принадлежат к некоторому типу (серьги, кольца, броши, браслеты), бывают выполнены из определенного материала, имеют некоторый вес и цену (включающую стоимость материалов и работы).

1. **Диаграмма вариантов использования**

Для того, чтобы более точно понять, как должна работать система, все чаще используется описание функциональности системы через варианты использования (Use Case или прецеденты).

Варианты использования это - описание последовательности действий, которые может осуществлять система в ответ на внешние воздействия пользователей или других программных систем. Варианты использования отражают функциональность системы с точки зрения получения значимого результата для пользователя, поэтому они точнее позволяют ранжировать функции по значимости получаемого результата.

Диаграмма вариантов использования состоит из актеров, для которых система производит действие и собственно действия Use Case, которое описывает то, что актер хочет получить от системы. Актер обозначается значком человечка, а Use Case - овалом. Дополнительно в диаграммы могут быть добавлены комментарии.

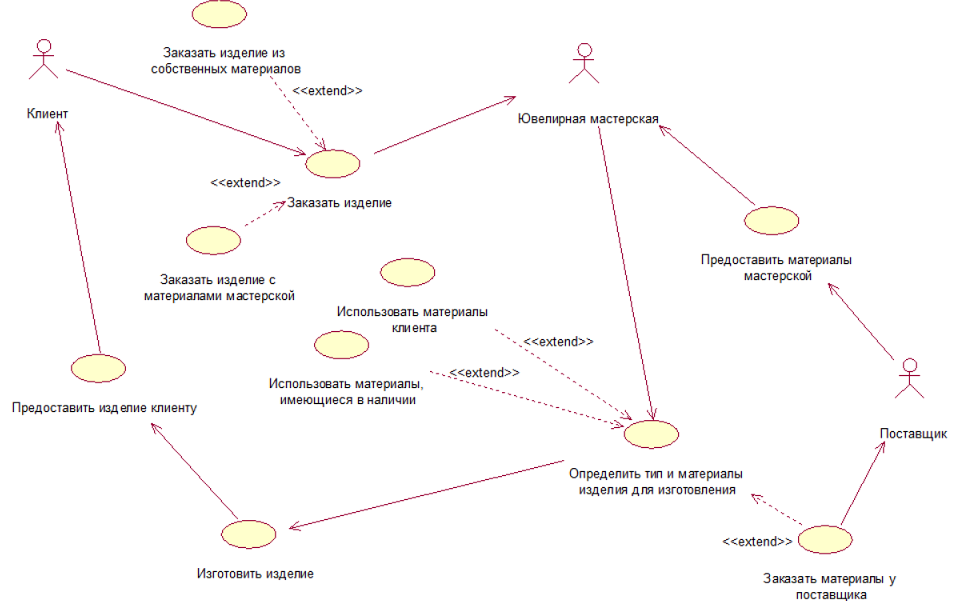


Рисунок 1.1 – диаграмма вариантов использования

Действующими лицами в данной системе являются «Клиент», «Ювелирная мастерская» и «Поставщик». На диаграмме показано взаимодействие между вариантами использования и действующими лицами, отражены требования к системе с точки зрения пользователя. Можно сказать, что варианты использования – это функции, выполняемые системой, а актеры – заинтересованные лица, инициирующие варианты использования, либо получающие от него информацию.

1. **Диаграмма вариантов использования**

Для того, чтобы более точно понять, как должна работать система, все чаще используется описание функциональности системы через варианты использования (Use Case или прецеденты).

Варианты использования это - описание последовательности действий, которые может осуществлять система в ответ на внешние воздействия пользователей или других программных систем. Варианты использования отражают функциональность системы с точки зрения получения значимого результата для пользователя, поэтому они точнее позволяют ранжировать функции по значимости получаемого результата.

Диаграмма вариантов использования состоит из актеров, для которых система производит действие и собственно действия Use Case, которое описывает то, что актер хочет получить от системы. Актер обозначается значком человечка, а Use Case - овалом. Дополнительно в диаграммы могут быть добавлены комментарии.

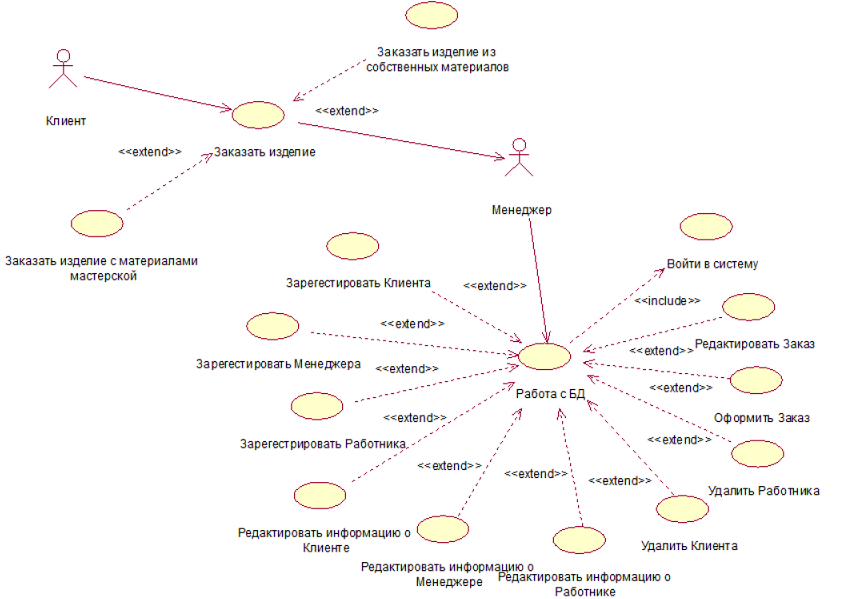


Рисунок 1.1 – диаграмма вариантов использования

Действующими лицами в данной системе являются «Клиент» и «Менеджер». На диаграмме показано взаимодействие между вариантами использования и действующими лицами, отражены требования к системе с точки зрения пользователя. Проектируемая система «Ювелирная мастерская» должна выполнять следующие действия: Заказать изделие с материалами мастерской, заказать изделие из собственных материалов, войти в систему, зарегистрировать клиента, зарегистрировать менеджера, зарегистрировать рабочего, редактировать информацию о клиенте, редактировать информацию о менеджере, редактировать информацию о работнике, удалить клиента, удалить работника, оформить заказ, редактировать заказ. Можно сказать, что варианты использования – это функции, выполняемые системой, а актеры – заинтересованные лица, инициирующие варианты использования, либо получающие от него информацию.

1. **Диаграмма последовательности**

Диаграмма последовательности является одной из разновидности диаграмм взаимодействия и предназначена для моделирования взаимодействия объектов Системы во времени, а также обмена сообщениями между ними. На ней изображаются объекты, которые непосредственно участвуют во взаимодействии, и сообщения, которыми эти объекты обмениваются.

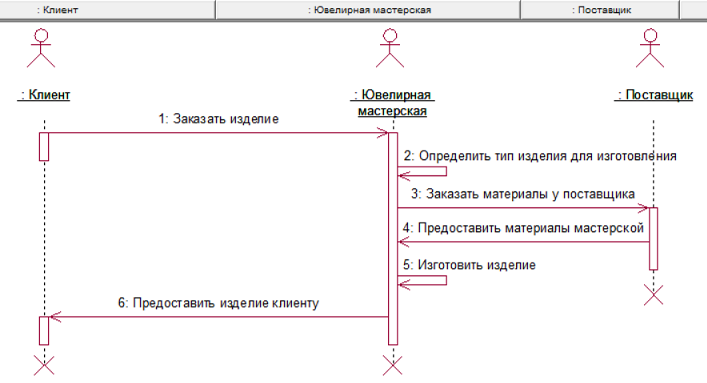


Рисунок 2.1 – Диаграмма последовательностей

На данной диаграмме изображена последовательность действий актеров данной предметной области. На первом шаге клиент заказывать изделие в мастерской, далее мастерская определяет тип изделия и заказывает необходимые изделия у поставщика (шаг 2 и 3). Четвёртым шагом поставщик предоставляет материалы мастерской. Мастерская изготавливает изделие и предоставляет его клиенту (шаг 5 и 6).

1. **Диаграмма последовательности**

Диаграмма последовательности является одной из разновидности диаграмм взаимодействия и предназначена для моделирования взаимодействия объектов Системы во времени, а также обмена сообщениями между ними. На ней изображаются объекты, которые непосредственно участвуют во взаимодействии, и сообщения, которыми эти объекты обмениваются.

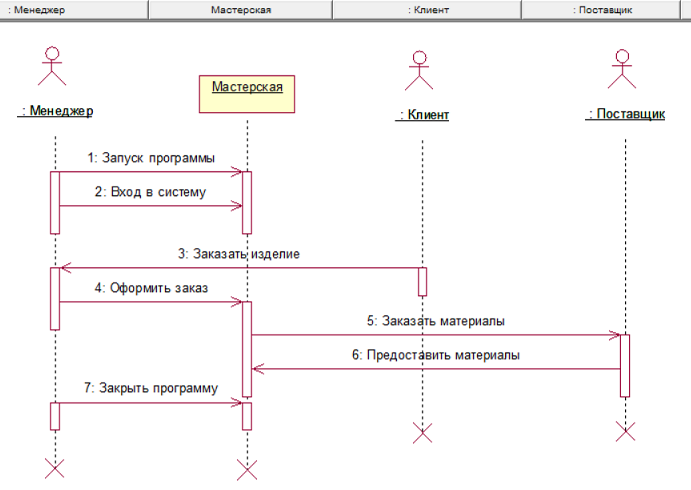


Рисунок 2.1 – Диаграмма последовательностей

На данной диаграмме изображена последовательность действий актеров данной предметной области. Первые два шага заключаются в том, что менеджер запускает программу и входит в систему. На третьем шаге клиент заказывать изделие, после чего менеджер оформляет заказ в систему. Далее заказываются необходимы материалы у поставщика (шаг 5). Шестым шагом поставщик предоставляет эти материалы. В конце менеджер закрывает программу.

1. **Диаграмма коопераций**

Диаграмма коопераций, в отличие от диаграммы последовательности, акцентирует внимание на организации объектов, принимающих участие во взаимодействии. Она представляется в виде графа, вершины которого – объекты, участвующие во взаимодействии, а дуги – связи, соединяющие объекты. Связи дополняются сообщениями, которые объекты принимают и посылают. Таким образом, диаграмма коопераций даёт визуальное представление о потоке управления в контексте структурной организации кооперирующихся объектов.

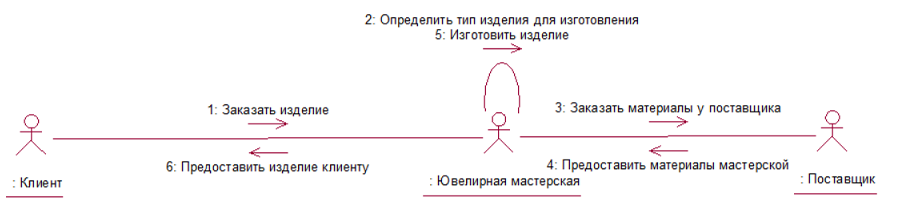


Рисунок 3.1 – диаграмма коопераций

1. **Диаграмма коопераций**

Диаграмма коопераций, в отличие от диаграммы последовательности, акцентирует внимание на организации объектов, принимающих участие во взаимодействии. Она представляется в виде графа, вершины которого – объекты, участвующие во взаимодействии, а дуги – связи, соединяющие объекты. Связи дополняются сообщениями, которые объекты принимают и посылают. Таким образом, диаграмма коопераций даёт визуальное представление о потоке управления в контексте структурной организации кооперирующихся объектов.

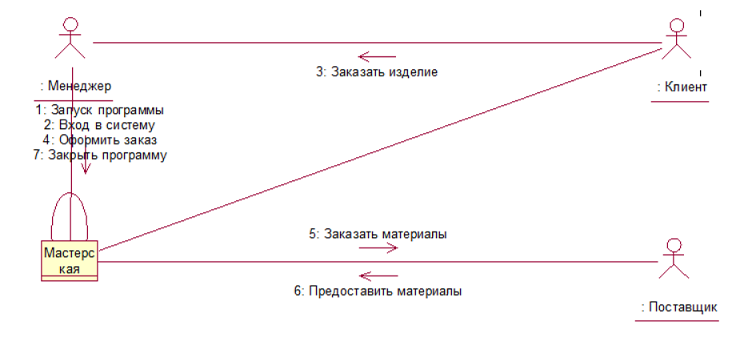


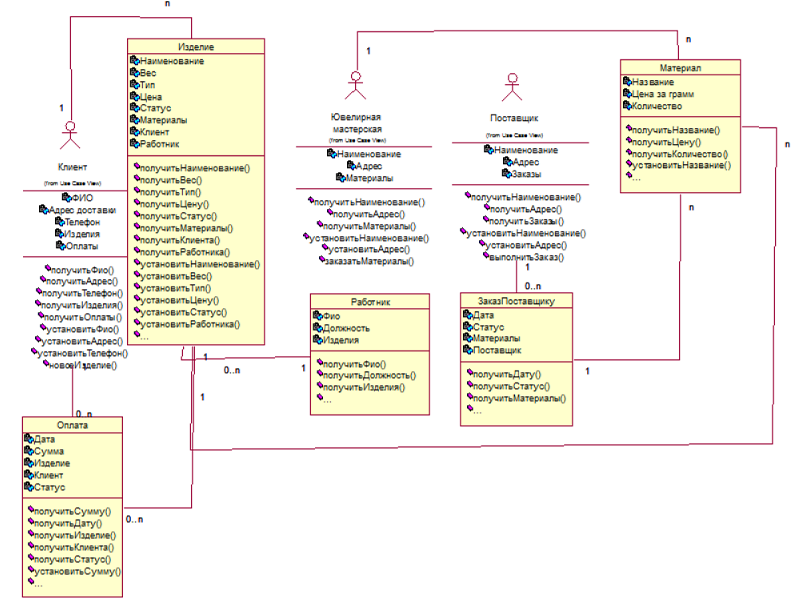
Рисунок 3.1 – диаграмма коопераций

1. **Диаграмма классов**

Диаграммы классов используются при моделировании ПС наиболее часто. Они являются одной из форм статического описания системы с точки зрения ее проектирования, показывая ее структуру. Диаграмма классов не отображает динамическое поведение объектов, изображенных на ней классов. На диаграммах классов показываются классы, интерфейсы и отношения между ними.

Атрибут – это именованное свойство класса, включающее описание множества значений, которые могут принимать экземпляры этого свойства.

Операции реализуют связанное с классом поведение, иными словами, абстракцию того, что позволено делать с объектом.

Рисунок 4.1 – Диаграмма классов

1. **Диаграмма деятельности**

Диаграмма деятельности – это технология, позволяющая описывать логику процедур, бизнес-процессы и потоки работ. Во многих случаях они напоминают блок-схемы, но принципиальная разница между диаграммами деятельности и нотацией блок-схем заключается в том, что первые поддерживают параллельное процессы.

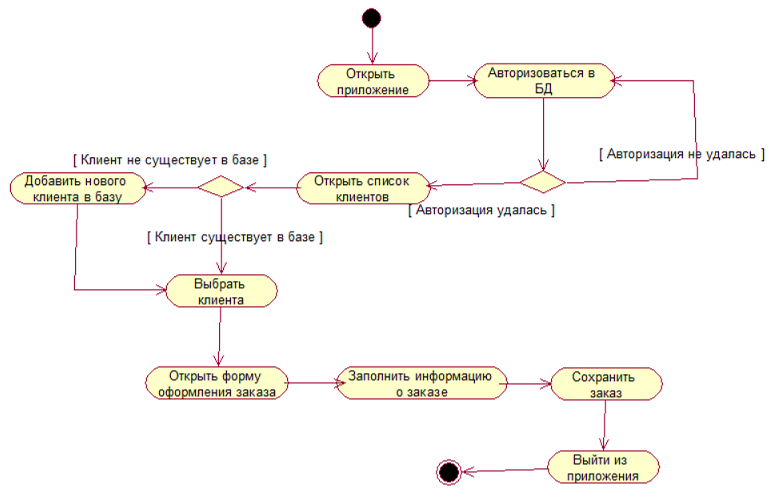


Рисунок 5.1 – Диаграмма деятельности

На данной диаграмме деятельности отображен процесс оформления заказа для определённого клиента.

1. **Диаграмма компонентов**

Диаграмма компонентов - статическая структурная диаграмма, показывает разбиение программной системы на структурные компоненты и связи (зависимости) между компонентами. В качестве физических компонентов могут выступать файлы, библиотеки, модули, исполняемые файлы, пакеты и т. п.

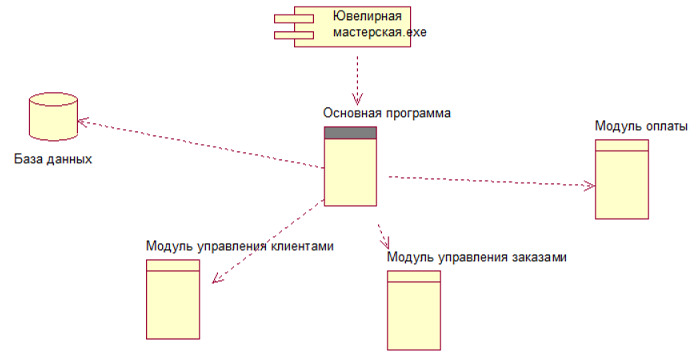


Рисунок 6.1 – Диаграмма компонентов