|  |  |  |
| --- | --- | --- |
| Дата | Содержание выполняемой работы | Подпись руководите ля |
| 12.05.2022 | **Практическая работа №14**  **Тема:** Комплексные системы разработки проектов.  **Наименование работы:** Разработка диаграмм деятельности и состояний.  **Цель:** Изучить нотации, применяемые при построении диаграммы деятельности и состояний, освоить их применение в процессе объектно-ориентированного анализа и проектирования.  **Задание 1.**  Изучил теоретический материал и примеры построения диаграмм деятельности, последовательно перейдя по ссылкам, законспектировал основные определения.  Диаграмма деятельности — это, по существу, блок-схема, которая показывает, как поток управления переходит от одной деятельности к другой, при этом внимание фиксируется на результате деятельности. Результат может привести к изменению состояния системы или возвращению некоторого значения.  Диаграмма деятельности отличается от традиционной блок-схемы более высоким уровнем абстракции. возможностью представления с помощью диаграмм деятельности управления параллельными потоками наряду с последовательным управлением.  Разработка диаграммы деятельности преследует цели: детализировать особенности алгоритмической и логической реализации выполняемых системой операций и прецедентов, выделить последовательные и параллельные потоки управления, подготовить детальную документацию для взаимодействия разработчиков системы с ее заказчиками и проектировщиками.  Графически диаграмма деятельности представляется в форме графа деятельности, вершинами которого являются состояния действия или состояния деятельности, а дугами - переходы от одного состояния действия/деятельности к другому. Каждая диаграмма деятельности должна иметь единственное начальное и единственное конечное состояния (на практике иногда можно видеть несколько конечных состояний на одной диаграмме, но это одно и тоже состояние, изображенное несколько раз для лучшей читабельности диаграммы). Саму диаграмму деятельности принято располагать таким образом, чтобы действия следовали сверху вниз. В этом случае начальное состояние будет изображаться в верхней части диаграммы, а конечное - в ее нижней части.  Создание Информационной Системы – сложный процесс, который можно представить как поэтапный спуск от общей концепции будущей ИС, через понимание ее логической структуры к наиболее детальным моделям, описывающим физическую реализацию. Диаграмма деятельности принадлежит к логической модели. В качестве графического представления для выделения основных функций Системы мы применяем диаграмму вариантов использования (use case).  Диаграмма вариантов использования дает нам представление ЧТО должна делать Система. На вопрос КАК мы можем ответить, используя диаграмму активности.  Для создания диаграммы деятельности используются следующие узлы:   1. Узел управления – это абстрактный узел действия, которое координирует потоки действий 2. Начальный узел деятельности является узлом управления, в котором начинается поток при вызове данной деятельности извне 3. Конечный узел деятельности является узлом управления, который останавливает все потоки данной диаграммы деятельности. На диаграмме может быть более одного конечного узла 4. Конечный узел потока является узлом управления, который завершает данный поток. На другие потоки и деятельность данной диаграммы это не влияет 5. Объект, над которым выполняются действия. Это не обязательный элемент диаграммы, но в некоторых случаях необходимо показать объект инициирующий выполнение действий, или являющийся результатом его.   Узел решения предназначен для определения правила ветвления и различных вариантов дальнейшего развития сценария.  В точку ветвления входит ровно один переход, а выходит - два или более. Для каждого исходящего перехода задается булевское выражение, которое вычисляется только один раз при входе в точку ветвления. Ни для каких двух исходящих переходов эти сторожевые условия не должны одновременно принимать значение "истина", иначе поток управления окажется неоднозначным. Желательно чтобы условия покрывали все возможные варианты, иначе поток остановится.  Переход - отношение между двумя состояниями, показывающее, что объект, находящийся в первом состоянии, должен выполнить некоторые действия и перейти во второе состояние. Когда действие или деятельность в некотором состоянии завершается, поток управления сразу переходит в следующее состояние действия или деятельности. Для описания этого потока и используются переходы, показывающие путь из одного состояния действия или деятельности в другое. В UML переход представляется простой линией со стрелкой.  Ветвления. Простые последовательные переходы встречаются наиболее часто, но их одних недостаточно для моделирования любого потока управления. Как и в блок-схему, в диаграмму деятельности может быть включено ветвление или множественный переход со сторожевыми условиями. Ветвление описывает различные пути выполнения в зависимости от значения некоторого булевского выражения. Графически точка ветвления представляется ромбом. В точку ветвления может входить ровно один переход, а выходить - два или более. Для каждого исходящего перехода задается булевское выражение, которое вычисляется только один раз при входе в точку ветвления. Ни для каких двух исходящих переходов сторожевые условия не должны одновременно принимать значение "истина", иначе поток управления окажется неоднозначным. Но эти условия должны покрывать все возможные варианты, иначе поток остановится.  Разделения и слияния. Простые и ветвящиеся последовательные переходы в диаграммах деятельности используются чаще всего. Однако часто возникает потребность изображения параллельных потоков, и это особенно характерно для моделирования бизнес-процессов. В UML для обозначения разделения и слияния таких параллельных потоков выполнения используется синхронизационная черта, которая рисуется в виде жирной вертикальной или горизонтальной линии. При этом разделение имеет один входящий переход и несколько выходящих, слияние, наоборот, имеет несколько входящих переходов и один выходящий.  Дорожки. При моделировании течения бизнес-процессов иногда бывает полезно разбить состояния деятельности на диаграммах деятельности на группы, каждая из которых представляет отдел компании, отвечающий за ту или иную работу. В UML такие группы называются дорожками, поскольку визуально каждая группа отделяется от соседней вертикальной черты, как плавательные дорожки в бассейне. Дорожки — это разновидность пакетов, описывающие связанную совокупность работ.  Каждой присутствующей на диаграмме дорожке присваивается уникальное имя. Никакой глубокой семантики дорожка не несет, разве что может отражать некоторую сущность реального мира. Каждая дорожка представляет сферу ответственности за часть всей работы, изображенной на диаграмме. На диаграмме деятельности, разбитой на дорожки, каждая деятельность принадлежит ровно одной дорожке, но переходы могут пересекать границы дорожек.  **Задание 2.**  Изучил теоретический материал и примеры построения диаграмм деятельности, последовательно перейдя по ссылкам, законспектировал основные определения.  Диаграмма состояний описывает процесс изменения состояний только одного класса, а точнее – одного экземпляра определенного класса, т. е. моделирует все возможные изменения в состоянии конкретного объекта. При этом изменение состояния объекта может быть вызвано внешними воздействиями со стороны других объектов или извне. Именно для описания реакции объекта на подобные внешние воздействия и используются диаграммы состояний.  Диаграмма состояний описывает возможные последовательности состояний и переходов, которые в совокупности характеризуют поведение элемента модели в течение его жизненного цикла. Диаграмма состояний представляет динамическое поведение сущностей, на основе спецификации их реакции на восприятие некоторых конкретных событий.  Диаграмма состояний по существу является графом специального вида, который представляет некоторый автомат. Понятие автомата в контексте UML обладает довольно специфической семантикой, основанной на теории автоматов. Вершинами этого графа являются состояния и некоторые другие типы элементов автомата, которые изображаются соответствующими графическими символами. Дуги графа служат для обозначения переходов из состояния в состояние. Диаграммы состояний могут быть вложены друг в друга, образуя вложенные диаграммы более детального представления отдельных элементов модели.  В общем случае автомат представляет динамические аспекты моделируемой системы в виде ориентированного графа, вершины которого соответствуют состояниям, а дуги – переходам. При этом поведение моделируется как последовательное перемещение по графу состояний от вершины к вершине по связывающим их дугам с учетом их ориентации.  Под состоянием понимается абстрактный метакласс, используемый для моделирования отдельной ситуации, в течение которой имеет место выполнение некоторого условия. Состояние может быть задано в виде набора конкретных значений атрибутов класса или объекта, при этом изменение их отдельных значений будет отражать изменение состояния моделируемого класса или объекта.  Метка действия указывает на обстоятельства или условия, при которых будет выполняться деятельность, определенная выражением действия:  entry – эта метка указывает на действие, специфицированное следующим за ней выражением действия, которое выполняется в момент входа в данное состояние (входное действие);  exit – эта метка указывает на действие, специфицированное следующим за ней выражением действия, которое выполняется в момент выхода из данного состояния (выходное действие);  do – эта метка специфицирует выполняющуюся деятельность, которая выполняется в течение всего времени, пока объект находится в данном состоянии, или до тех пор, пока не закончится вычисление, специфицированное следующим за ней выражением действия. В последнем случае при завершении события генерируется соответствующий результат;  include – эта метка используется для обращения к подавтомату, при этом следующее за ней выражение действия содержит имя этого подавтомата.  Начальное состояние представляет собой частный случай состояния, которое не содержит никаких внутренних действий. В этом состоянии находится объект по умолчанию в начальный момент времени. Оно служит для указания на диаграмме состояний графической области, от которой начинается процесс изменения состояний.  Конечное состояние представляет собой частный случай состояния, которое также не содержит никаких внутренних действий. В этом состоянии будет находиться объект по умолчанию после завершения работы автомата в конечный момент времени.  Простой переход представляет собой отношение между двумя последовательными состояниями, которое указывает на факт смены одного состояния другим. Пребывание моделируемого объекта в первом состоянии может сопровождаться выполнением некоторых действий, а переход во второе состояние будет возможен после завершения этих действий, а также после удовлетворения некоторых дополнительных условий. В этом случае говорят, что переход срабатывает, Или происходит срабатывание перехода.  Событие представляет собой спецификацию некоторого факта, имеющего место в пространстве и во времени. Про события говорят, что они «происходят», при этом отдельные события должны быть упорядочены во времени. После наступления некоторого события нельзя уже вернуться к предыдущим событиям, если такая возможность не предусмотрена явно в модели.  Сторожевое условие, если оно есть, всегда записывается в прямых скобках после события и представляет собой некоторое булевское выражение. Если сторожевое условие принимает значение «истина», то соответствующий переход может сработать, в результате чего объект перейдет в целевое состояние. Если же сторожевое условие принимает значение «ложь», то переход не может сработать, и при отсутствии других переходов объект не может перейти в целевое состояние по этому переходу.  **Задание 3.**  Построил диаграмму деятельности и состояний для каждого прецедента, присутствующего на диаграмме прецедентов, разработанной при выполнении практической работы 13. Диаграмму деятельности и состояний оформил в Приложении 14.1. |  |