**Государственное бюджетное профессиональное образовательное учреждение города Москвы "Московский автомобильно-дорожный колледж им. А.А. Николаева"**

**Ответы**

**на контрольные вопросы**

**по инструментальным средствам разработки программного обеспечения**

**на тему:**

**Диаграммы вариантов использования**

**Выполнил студент группы 2ИП1**

**Копачински Ольга**

1. **Что такое нефункциональные требования?**

Нефункциональные требования – это требования, которые определяют характеристики и качественные свойства системы или продукта, а не их функциональность. Они описывают, каким должно быть поведение системы в рамках нефункциональных аспектов, таких как производительность, безопасность, надежность, удобство использования и совместимость.

1. **Как нефункциональные требования отображаются на диаграммах прецедентов?**

* Использование актеров: Нефункциональные требования могут быть связаны с актерами, чтобы показать какие-то специфические навыки или способы взаимодействия с системой.
* Использование расширений: Нефункциональные требования могут быть отражены в виде расширений или альтернативных потоков выполнения.

1. **Какие способы изображения акторов вы знаете?**

На диаграмме вариантов использования акторы обычно изображаются в виде символа человека или некоторого другого объекта.

1. **В какие отношения могут вступать акторы между собой?**

Отношения, которые могут вступать акторы между собой на диаграмме вариантов использования, включают:

* Отношение "ассоциации": показывает взаимодействие между актором и вариантом использования. Подразумевается, что актор использует определенный вариант использования.
* Отношение "включение": показывает, что один вариант использования включает в себя другой вариант использования как свою составную часть.
* Отношение "расширение": демонстрирует, что один вариант использования может быть расширен другим вариантом использования в случае определенных условий. Расширенный вариант использования является необязательным и может быть выполнен только при наступлении определенных условий.
* Отношение "общий актор": указывает на то, что несколько акторов имеют общие характеристики, но один актор является более общим, чем другой. Например, акторы "клиент" и "администратор" могут быть подразумеваемыми под более общим актором "пользователь".

1. **В чем состоит смысл отношений включения?**

Смысл отношения включения состоит в том, чтобы показать, что один вариант использования включает в себя функциональность другого варианта использования. Это означает, что один вариант использования является частью другого варианта использования и может использовать его функциональность, чтобы расширить свою функциональность или улучшить его работу.

1. **В чем состоит смысл отношений расширения?**

Смысл отношений расширения состоит в возможности расширения одного варианта использования другим вариантом при определенных условиях. Расширенный вариант использования является необязательным и может быть выполнен только в случае выполнения этих условий. Это позволяет уточнить функциональность и добавить дополнительные возможности к базовому варианту использования.

1. **Что такое точка расширения?**

**Точки расширения**— это особый элемент на диаграмме вариантов использования (use case diagram), который позволяет представить возможность расширения функциональности системы путем внедрения дополнительных вариантов использования (use cases) без изменения основной логики приложения.

1. **Перечислите известные вам причины использования прецедентов.**

* Представление функциональных возможностей системы или ее частей.
* Описание конкретных задач, которые система должна выполнять для достижения цели.
* Представление сценариев использования системы с точки зрения пользователей.
* Описание возможных вариантов поведения системы в определенных условиях, таких как ошибки ввода или некорректные данные.
* Связывание вариантов использования с акторами, чтобы показать, какие действия могут выполнять акторы и как система должна на них реагировать.

1. **Как прецеденты применяют в прямом и обратном проектировании?**

В прямом проектировании прецеденты используются для определения функциональных возможностей системы и ее частей. Они помогают описать, какие конкретные задачи система должна выполнять для достижения своей цели. Прецеденты предоставляют сценарии использования системы, представляя ее функциональность с точки зрения пользователей.

В обратном проектировании прецеденты применяются для анализа уже существующей системы. Они помогают понять, как система используется, какие задачи она выполняет и как она взаимодействует с акторами. Прецеденты в обратном проектировании используются для восстановления и документирования функциональности системы.

1. **Назовите сходства и различия диаграмм прецедентов и контекстных диаграмм?**

Диаграммы прецедентов (Use Case Diagrams) и контекстные диаграммы являются инструментами моделирования в области анализа и проектирования программного обеспечения. Вот сходства и различия между этими двумя типами диаграмм:

Сходства между диаграммой прецедентов и контекстной диаграммой:

* Оба типа диаграмм используются для визуализации взаимодействий между актерами (пользователями) и системой.
* Оба типа диаграмм помогают в понимании и определении функциональности системы.
* В обоих диаграммах обычно используются прямоугольники для представления актеров и внешних систем.

Различия между диаграммой прецедентов и контекстной диаграммой:

* Контекстная диаграмма фокусируется на общем представлении системы и ее внешних сущностей, тогда как диаграмма прецедентов углубляется в детали внутренних функций системы.
* Диаграмма прецедентов показывает взаимодействия между актерами и конкретными функциональными единицами системы, в то время как контекстная диаграмма описывает только высокоуровневые действия и взаимодействия актеров без уточнения деталей функциональности.
* При построении контекстной диаграммы акцент делается на разграничении системы от внешних сущностей, тогда как в диаграмме прецедентов акцент делается на выявлении основных функциональных процессов и ролей актеров.

1. **О каких вариантах (прецедентах, сценариях) использования дают представление UseCaseDiagrams**

Используется для описания функциональных требований к системе, ее возможных сценариев использования и взаимодействия системы с внешними сущностями (акторами).

1. **Назовите сходства и различия экторов и внешних сущностей.**

Одним из сходств между акторами и внешними сущностями является то, что оба представляют внешние элементы взаимодействия с системой. Они могут инициировать действия и взаимодействовать с системой для достижения определенных целей.

Однако, есть и различия между акторами и внешними сущностями. В отличие от внешних сущностей, акторы могут иметь свои собственные цели, задачи и взаимодействия с системой. Они могут быть субъектами или участниками процесса, в то время как внешние сущности могут быть просто источниками входных данных или получателями выходных данных.

1. **Назовите сходства и различия прецедентов (на UseCaseDiagram ) и процессов (на ДПД).**

Оба представляют собой графические инструменты моделирования для описания работы системы.

Используются для описания функциональных требований и взаимодействия системы с внешними сущностями.

Различия прецедентов и процессов:

Прецеденты описывают возможные сценарии использования системы, позволяя идентифицировать акторов и их взаимодействие с системой. Процессы, с другой стороны, описывают последовательность шагов и действий, которые необходимо выполнить для достижения определенной цели.

Прецеденты ориентированы на пользовательские действия и взаимодействие с системой, в то время как процессы могут охватывать как пользовательские действия, так и внутренние операции системы.

1. **Для чего используются диаграммы прецедентов (вариантов использования)?**

Диаграммы прецедентов используются для моделирования и визуализации функциональных возможностей системы или ее частей. Они помогают описать задачи, которые система должна выполнять для достижения своих целей, и представляют сценарии использования системы с точки зрения ее пользователей. Кроме того, они позволяют описать возможные варианты выполнения для каждого варианта использования, включая описание ошибок ввода или некорректных данных. Диаграммы прецедентов также позволяют связывать варианты использования с акторами, представляющими внешние сущности, взаимодействующие с системой, и показывают, какие действия могут выполнять акторы и как система должна на них реагировать. В целом, диаграммы прецедентов помогают лучше понять функциональные требования системы и облегчают коммуникацию между разработчиками и пользователем.

1. **Что отображает (представляет) «прецедент» на Диаграмме прецедентов?**

**Диаграмма прецедентов** (диаграмма вариантов использования) в UML — диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Суть диаграммы вариантов использования заключается в том, что она помогает лучше понять требования к системе и определить ее функциональные возможности.

1. **Что такое «эктор» (актер, действующее лицо), что он отображает на диаграмме прецедентов?**

Актор (Actor) на диаграмме прецедентов представляет внешнюю сущность, которая взаимодействует с системой. Актором может быть любым человеком, группой людей, другой системой или даже другим компонентом системы.

На диаграмме прецедентов акторы обычно изображаются в виде символа человека или другого объекта. Они связываются с вариантами использования системы, которые могут инициировать. Связь между актором и вариантом использования показывает, что актор может выполнить определенные действия, которые система должна обработать.

1. **Назовите основные типы «экторов».**

Основные типы акторов на диаграмме прецедентов:

* *Пользователь - человек, который использует систему для достижения своих целей.*
* *Группа пользователей - несколько людей, работающих вместе или имеющих общий доступ к системе.*
* *Внешняя система - другая система, которая взаимодействует с данной системой.*
* *Компонент системы - часть самой системы, которая выполняет определенные функции.*

1. **Какие типы отношений (связей) между экторами и прецедентами используются на диаграммах прецедентов?**

На диаграммах прецедентов используются следующие типы отношений (связей) между акторами и прецедентами:

* Ассоциация (Association) - отношение, показывающее, что актор связан с одним или несколькими вариантами использования системы. Ассоциация обозначается стрелкой, указывающей на вариант использования.
* Включение (Include) - отношение, показывающее, что один вариант использования может включать другой вариант использования. Включение обозначается пунктирной стрелкой, указывающей на включаемый вариант использования.
* Расширение (Extend) - отношение, показывающее, что один вариант использования может расширять другой вариант использования. Расширение обозначается пунктирной стрелкой с надписью "extends", указывающей на расширяемый вариант использования.
* Обобщение (Generalization) - отношение, показывающее, что один актор является обобщением другого актора. Обобщение обозначается стрелкой с надписью "generalizes", указывающей на более общего актора.

1. **Почему (кроме созвучия английскому actors)экторчасто переводится как актер**

Перевод эктора как актера основан на созвучии с английским словом "actors". Однако, в контексте использования в информатике и системном анализе, эктор обычно относится к лицам или объектам, которые выполняют определенные роли или функции в системе, а не к профессиональным актерам.

1. **Какие еще варианты перевода actors на русский вам известны?**

К другим вариантам перевода actors на русский язык относятся "исполнители", "участники", "роли" или "функции".

1. **Совпадает ли понятие «эктор» с понятием «физический пользователь»?**

Понятие "эктор" не совпадает полностью с понятием "физический пользователь". Физический пользователь - это конкретная личность или лицо, которое использует систему. В то время как эктор более абстрактная роль или функция, которую пользователь может выполнять в системе.

1. **На какие 3 типа можно подразделять экторов?**

* Человеки: Физические пользователи или конкретные люди, которые взаимодействуют с системой.
* Машины: Аппаратное или программное обеспечение, которое взаимодействует с системой.
* Внешние системы Внешние организации или системы, с которыми система обменивается информацией или взаимодействует.

1. **Что представляет (описывает, отображает) прецедент?**

Прецедент представляет собой ситуацию или событие, которое описывает, как система взаимодействует с пользователем или другими сторонами. Он определяет, что должно произойти, когда определенные действия выполняются или определенные события возникают. Прецедент содержит информацию о вводе (входных данных), действиях (процессе) и результате (выходных данных), которые связаны с конкретной функцией системы.

1. **Какие типы связей (отношений) допускаются между экторами?**

* Связь "ассоциация" (association) - показывает взаимодействие между акторами и прецедентами. Она показывает, что актор использует или взаимодействует с определенным прецедентом.
* Связь "включение" (include) - указывает, что один вариант использования (прецедент) использует функциональность другого варианта использования. Это отношение показывает, что один вариант использования является составной частью другого варианта использования.
* Связь "расширение" (extend) - указывает на то, что один вариант использования может быть расширен другим вариантом использования при определенных условиях. Это отношение показывает, что расширенный вариант использования является необязательным и может быть выполнен только при определенных условиях.
* Связь "общий актор" (generalization) - указывает, что несколько акторов имеют общие характеристики, но один актор является более общим, чем другой. Например, на диаграмме может быть представлено, что акторы "клиент" и "администратор" являются более общим актором "пользователь".

1. **Почему не рекомендуется подробная детализация диаграмм прецедентов?**

Подробная детализация диаграмм прецедентов не рекомендуется, потому что она может привести к перегруженности и излишней сложности диаграммы. Отражать все возможные варианты использования системы на одной диаграмме просто невозможно. Более того, такая подробная детализация может усложнить понимание диаграммы и затруднить коммуникацию между разработчиками и заказчиками.