

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Пермский государственный национальный
исследовательский университет»**

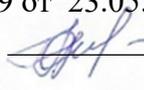
Колледж профессионального образования

**МЕТОДЫ И СРЕДСТВА ПРОЕКТИРОВАНИЯ
ИНФОРМАЦИОННЫХ СИСТЕМ**

Методические рекомендации

для практических работ по изучению дисциплины
для студентов Колледжа профессионального образования
специальности

09.02.04 Информационные системы (по отраслям)

Утверждено на заседании ПЦК
Информационных технологий
Протокол № 9 от 23.05.2018
председатель  Н.А. Серебрякова

Пермь 2018

Составитель:

Серебрякова Н.А., преподаватель ФГБОУ Пермский государственный национальный исследовательский университет, Колледж профессионального образования

Методы и средства проектирования информационных систем: методические указания для практических работ по изучению дисциплины для студентов Колледжа профессионального образования специальности 09.02.03 **09.02.04 Информационные системы (по отраслям)** / сост. Н.А.Серебрякова; Колледж проф. образ. ПГНИУ. – Пермь, 2019. -14 с.

Методические указания **Методы и средства проектирования информационных систем** разработаны на основе требований Федерального государственного стандарта среднего профессионального образования по специальности **09.02.04 Информационные системы (по отраслям)** для оказания помощи студентам по дисциплине **Методы и средства проектирования информационных систем** Предназначены для студентов Колледжа профессионального образования ПГНИУ специальности **09.02.04 Информационные системы (по отраслям)** (СПО) всех форм обучения.

ОГЛАВЛЕНИЕ

1	ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	4
2	ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ	6
3	МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНЫХ ЗАДАНИЙ ПО МДК 01.02 «МЕТОДЫ И СРЕДСТВА ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ»	7
3.1	РАЗДЕЛ 1. ВВЕДЕНИЕ В ПРОЕКТИРОВАНИЕ ПО	7
3.2	РАЗДЕЛ 2. СТРУКТУРНЫЕ МЕТОДЫ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ	8
3.3	РАЗДЕЛ 3. МЕТОД СУЩНОСТЬ-СВЯЗЬ (ER)	19
3.4	РАЗДЕЛ 4. ВИЗУАЛЬНОЕ МОДЕЛИРОВАНИЕ НА ЯЗЫКЕ UML	20
3.5	РАЗДЕЛ 5. ТЕХНОЛОГИИ СОЗДАНИЯ ПО	23
3.6	РАЗДЕЛ 6. ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	24
4	ЛИТЕРАТУРА ДЛЯ УЧАЩЕГОСЯ	28

1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Данные методические рекомендации направлены на реализацию самостоятельной работы по Методы и средства проектирования информационных систем для студентов СПО по специальности 09.02.04 «Информационные системы (по отраслям)».

Самостоятельная работа учащихся является одним из основных методов приобретения и углубления знаний, познания общественной практики. Главной задачей самостоятельной работы является развитие общих и профессиональных компетенций, умений приобретать научные знания путем личных поисков, формирование активного интереса к творческому самостоятельному подходу в работе.

Самостоятельная работа складывается из изучения конспектов лекций, учебной и специальной литературы, как основной, так и дополнительной, нормативного материала, выполнения практических заданий, подготовки сообщений и презентаций по темам.

Методические рекомендации по выполнению самостоятельной работы разработаны в соответствии с программой профессионального модуля ПМ.01 «Эксплуатация и модификация информационных систем» для учащихся СПО по специальности 09.02.04 «Информационные системы (по отраслям)».

В ходе изучения дисциплины ставиться задача формирования следующих компетенций:

общие:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес

ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития

ОК 5. Использовать информационно - коммуникационные технологии в профессиональной деятельности

ОК 6. Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями

ОК 7. Брать на себя ответственность за работу членов команды (подчинённых), результат выполнения заданий

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности

профессиональные:

ПК1.1. Собирать данные для анализа использования и функционирования информационной системы, участвовать в составлении отчетной документации, принимать участие в разработке проектной документации на модификацию информационной системы.

ПК1.2. Взаимодействовать со специалистами смежного профиля при разработке методов, средств и технологий применения объектов профессиональной деятельности.

ПК1.3. Производить модификацию отдельных модулей информационной системы в соответствии с рабочим заданием, документировать произведенные изменения.

ПК1.4. Участвовать в экспериментальном тестировании информационной системы на этапе опытной эксплуатации, фиксировать выявленные ошибки кодирования в разрабатываемых модулях информационной системы.

ПК1.5. Разрабатывать фрагменты документации по эксплуатации информационной системы.

ПК1.6. Участвовать в оценке качества и экономической эффективности информационной системы.

ПК1.7. Производить инсталляцию и настройку информационной системы в рамках своей компетенции, документировать результаты работ.

ПК1.8. Консультировать пользователей информационной системы и разрабатывать фрагменты методики обучения пользователей информационной системы.

ПК1.9. Выполнять регламенты по обновлению, техническому сопровождению и восстановлению данных информационной системы, работать с технической документацией.

ПК1.10. Обеспечивать организацию доступа пользователей информационной системы в рамках своей компетенции.

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО
ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНЫХ ЗАДАНИЙ ПО МДК 01.02
«МЕТОДЫ И СРЕДСТВА ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ
СИСТЕМ»**

РАЗДЕЛ 1. ВВЕДЕНИЕ В ПРОЕКТИРОВАНИЕ ПО

Тема 1.1. Введение в проектирование ПО

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников.

Ключевым для данной темы являются понятия: проектирование программного обеспечения, проект программного обеспечения, программная инженерия, технология программирования.

Студенты должны изучить данные понятия, разобраться в особенностях проектирования программного обеспечения, изучить предпосылки создания дисциплины программная инженерия.

Тема 1.2. Методы и технологии проектирования

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников.

Ключевым для данной темы являются понятия: декомпозиция системы, структурный и объектно-ориентированный подход к разработке программного обеспечения.

Студенты должны изучить данные подходы. Знать понятие Технология проектирования ПО, структурные методы проектирования ПО: IDEF0, IDEF3, DFD. Особенности объектно-ориентированного метода проектирования ПО.

Тема 1.3. Средства проектирования ПО

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников.

Ключевым для данной темы являются понятия: средства проектирования, CASE-средства, классификация CASE-средств.

Студенты должны иметь представления о CASE-средствах: ERwin Process Modeler, StarUML, CASE.Аналитик, Rational Rose, ARIS и пр.

Методические рекомендации по подготовке сообщения

Целью данной самостоятельной работы является знакомство с характеристиками современных CASE-средств.

Студентам выдается задание: Используя ресурсы Интернет и предложенную литературу выполнить анализ существующих CASE-средств по критериям, представленным ниже:

- Название
- Фирма и страна разработчик
- Год последнего обновления
- Назначение
- Функциональные возможности
- Поддерживаемые диаграммы/модели
- Цена/Лицензия распространения

Ответы по данным критериям представить в презентации Power Point. Задание выполняется в микрогруппе по 2-3 человека. Каждая микрогруппа выбирает для анализа одно из CASE-средств, представленных ниже или предлагает свое CASE-средство. Выбранный вариант согласовывается с преподавателем. Варианты CASE-средств:

1	ERwin Data Modeler
2	ERwin Process Modeler

3	Альтернатива 1 BPWin
4	Microsoft Visio
5	CASE.Аналитик
6	ARIS
7	StarUML
8	Silverrun
9	Rational Rose

2.1 РАЗДЕЛ 2. СТРУКТУРНЫЕ МЕТОДЫ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

Тема 2.1. Проведение предпроектного обследования предприятия

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников.

Ключевым для данной темы являются понятия предпроектного обследования предприятия.

Студенты должны знать технологию проведения предпроектного обследования предприятия и формирования отчета по предпроектному обследованию предприятия.

Тема 2.2. Структурные методы проектирования информационных систем

Методические рекомендации по подготовке к практической работе и подготовке презентации и сообщения

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников, выполнение самостоятельного практического задания по предложенным темам.

Ключевым для данной темы являются понятия структурных методов разработки модели бизнес-процессов предприятия.

Студенты должны знать методологии построения моделей IDEF0, IDEF3, DFD. Закрепление навыков проектирования моделей вышеперечисленными

методами осуществляется при выполнении самостоятельного практического задания:

Самостоятельное практическое задание:

Для выбранной предметной области спроектировать модели IDEF0, IDEF3, DFD.

Цель выполнения задания - закрепление практических навыков проектирования IDEF0, IDEF3, DFD-моделей.

Результатом выполнения заданий является полная модель бизнес-процессов предприятия, разработанная студентами.

Задание:

1. С целью анализа деятельности выбранного предприятия (варианты предметных областей представлены ниже) создайте IDEF0-модель бизнес-процессов предприятия, включающую:

- Контекстную диаграмму;
- Диаграммы декомпозиции (первого и второго уровня, от 3-х до 5 работ в каждой диаграмме декомпозиции);
- Диаграмму дерева узлов;

2. Создайте IDEF3-модель бизнес-процессов предприятия.

3. Создайте DFD-модель бизнес-процессов предприятия.

Задание выполняется в микрогруппе по 2-3 человека. Каждая микрогруппа выбирает один из вариантов предложенных предметных областей. Выбор согласовывается с преподавателем.

Сохранить созданные модели в виде картинок в документе Word. Сформированный документ представить преподавателю для проверки.

Варианты предметных областей:

1. Система автоматизации гостиницы
2. Система автоматизации ресторана
3. CRM-система
4. Библиотечная система

5. Система автоматизации автобусного парка
6. Система автоматизации парковки
7. Система автоматизации проката автомобиля
8. Система кредитования банка
9. Система автоматизации турбазы
10. Система автоматизации салона красоты
11. Система автоматизации фитнес-центра

Методические рекомендации по выполнению самостоятельного практического задания:

IDEF0

IDEF0 сочетает в себе небольшую по объему графическую нотацию – блоки и стрелки со строгими и четко определенными рекомендациями их проектирования. Процесс моделирования системы в IDEF0 начинается с создания контекстной диаграммы — диаграммы наиболее абстрактного уровня описания системы в целом, содержащей определение субъекта моделирования, цели и точки зрения на модель.

Под субъектом понимается сама система, при этом необходимо точно установить, что входит в систему, а что лежит за ее пределами, другими словами, определить, что будет в дальнейшем рассматриваться как компоненты системы, а что как внешнее воздействие. В начале необходимо определить область моделирования.

При формулировании области необходимо учитывать два компонента — широту и глубину. Широта подразумевает определение границ модели — что будет рассматриваться внутри системы, а что снаружи. Глубина определяет, на каком уровне детализации модель является завершенной. При определении глубины системы необходимо помнить об ограничениях времени — трудоемкость построения модели растет в геометрической прогрессии с увеличением глубины декомпозиции. После определения границ модели

предполагается, что новые объекты не должны вноситься в моделируемую систему.

Цель моделирования

Цель моделирования определяется из ответов на следующие вопросы:

Почему этот процесс должен быть смоделирован?

Что должна показывать модель?

Что может получить клиент?

Точка зрения (Viewpoint).

Под точкой зрения понимается перспектива, с которой наблюдалась система при построении модели. Хотя при построении модели учитываются мнения различных людей, все они должны придерживаться единой точки зрения на модель. Точка зрения должна соответствовать цели и границам моделирования. Как правило, выбирается точка зрения человека, ответственного за моделируемую работу в целом.

IDEF0-модель предполагает наличие четко сформулированной цели, единственного субъекта моделирования и одной точки зрения. Для внесения области, цели и точки зрения в модели IDEF0 в VPwin следует выбрать пункт меню Model/Model Properties, вызывающий диалог Model Properties.

Результат описания модели можно получить в отчете Model Report. Диалог настройки отчета по модели вызывается из пункта меню Tools/Reports/Model Report. В диалоге настройки следует выбрать необходимые поля, при этом автоматически отображается очередность вывода информации в отчет.

Основу методологии IDEF0 составляет графический язык описания бизнес-процессов. Модель в нотации IDEF0 представляет собой совокупность иерархически упорядоченных и взаимосвязанных диаграмм. Каждая диаграмма является единицей описания системы и располагается на отдельном листе.

Модель может содержать четыре типа диаграмм:

- контекстную диаграмму (в каждой модели может быть только одна);
- диаграммы декомпозиции;
- диаграммы дерева узлов;

- диаграммы только для экспозиции (FEO).

Контекстная диаграмма является вершиной древовидной структуры диаграмм и представляет собой самое общее описание системы и ее взаимодействия с внешней средой. После описания системы в целом проводится разбиение ее на крупные фрагменты. Этот процесс называется функциональной декомпозицией, а диаграммы, которые описывают каждый фрагмент и взаимодействие фрагментов, называются диаграммами декомпозиции.

После декомпозиции контекстной диаграммы проводится декомпозиция каждого большого фрагмента системы на более мелкие и так далее, до достижения нужного уровня подробности описания. После каждого сеанса декомпозиции проводятся сеансы экспертизы — эксперты предметной области указывают на соответствие реальных бизнес-процессов созданным диаграммам.

Диаграмма дерева узлов показывает иерархическую зависимость работ, но не взаимосвязи между работами. Диаграмм деревьев узлов может быть в модели сколь угодно много, поскольку дерево может быть построено на произвольную глубину и не обязательно с корня.

Диаграммы для экспозиции (FEO) строятся для иллюстрации отдельных фрагментов модели, для иллюстрации альтернативной точки зрения, либо для специальных целей.

Работы (Activity)

обозначают поименованные процессы, функции или задачи, которые происходят в течение определенного времени и имеют распознаваемые результаты. Работы изображаются в виде прямоугольников. Все работы должны быть названы и определены. Имя работы должно быть выражено отглагольным существительным, обозначающим действие (например, "Деятельность компании", "Прием заказа" и т.д.).

Для внесения имени работы следует щелкнуть по работе правой кнопкой мыши, выбрать в меню Name Editor и в появившемся диалоге внести имя работы. Для описания других свойств работы служит диалог Activity Properties.

Диаграммы декомпозиции содержат родственные работы, т. е. дочерние работы, имеющие общую родительскую работу. Для создания диаграммы декомпозиции следует щелкнуть по кнопке ■ на панели инструментов.

Возникает диалог Activity Box Count, в котором следует указать нотацию новой диаграммы и количество работ на ней. Остановимся пока на нотации IDEF0 и щелкнем на ОК. Появляется диаграмма декомпозиции. Допустимый интервал числа работ — 2-8. Декомпозировать работу на одну работу не имеет смысла: диаграммы с количеством работ более восьми получаются перенасыщенными и плохо читаются. Для обеспечения наглядности и лучшего понимания моделируемых процессов рекомендуется использовать от трех до шести блоков на одной диаграмме.

Если оказывается, что количество работ недостаточно, то работу можно добавить в диаграмму, щелкнув сначала по кнопке на палитре инструментов, а затем по свободному месту на диаграмме.

Работы на диаграммах декомпозиции обычно располагаются по диагонали от левого верхнего угла к правому нижнему.

Такой порядок называется порядком доминирования. Согласно этому принципу расположения в левом верхнем углу помещается самая важная работа или работа, выполняемая по времени первой. Далее вправо вниз располагаются менее важные или выполняемые позже работы. Такое размещение облегчает чтение диаграмм, кроме того, на нем основывается понятие взаимосвязей работ.

Каждая из работ на диаграмме декомпозиции может быть в свою очередь декомпозирована. На диаграмме декомпозиции работы нумеруются автоматически слева направо. Номер работы показывается в правом нижнем углу. В левом верхнем углу изображается небольшая диагональная черта, которая показывает, что данная работа не была декомпозирована.

Стрелки(Arrow)

описывают взаимодействие работ и представляют собой некую информацию, выраженную существительными.(Например, "Звонки клиентов", "Правила и процедуры", "Бухгалтерская система".)

В IDEF0 различают пять типов стрелок:

Вход(Input) — материал или информация, которые используются или преобразуются работой для получения результата (выхода). Допускается, что работа может не иметь ни одной стрелки входа. Стрелка входа рисуется как входящая в левую грань работы. При описании технологических процессов (для этого и был придуман IDEF0) не возникает проблем определения входов. При моделировании ИС, когда стрелками являются не физические объекты, а данные, не все так очевидно. Например, при "Приеме пациента" карта пациента может быть и на входе и на выходе, между тем качество этих данных меняется. Другими словами, в нашем примере для того, чтобы оправдать свое назначение, стрелки входа и выхода должны быть точно определены с тем, чтобы указать на то, что данные действительно были переработаны (например, на выходе — "Заполненная карта пациента"). Очень часто сложно определить, являются ли данные входом или управлением. В этом случае подсказкой может служить информация о том, перерабатываются/изменяются ли данные в работе или нет. Если изменяются, то, скорее всего, это вход, если нет — управление.

Управление(Control) — правила, стратегии, процедуры или стандарты, которыми руководствуется работа. Каждая работа должна иметь хотя бы одну стрелку управления. Стрелка управления рисуется как входящая в верхнюю грань работы. Управление влияет на работу, но не преобразуется работой. Если цель работы — изменить процедуру или стратегию, то такая процедура или стратегия будет для работы входом. В случае возникновения неопределенности в статусе стрелки (управление или вход) рекомендуется рисовать стрелку управления.

Выход(Output) — материал или информация, которые производятся работой. Каждая работа должна иметь хотя бы одну стрелку выхода. Работа без результата не имеет смысла и не должна моделироваться. Стрелка выхода рисуется как исходящая из правой грани работы.

Механизм(Mechanism) — ресурсы, которые выполняют работу, например персонал предприятия, станки, устройства и т. д. Стрелка механизма рисуется

как входящая в нижнюю грань работы. По усмотрению аналитика стрелки механизма могут не изображаться в модели.

Вызов(Call) — специальная стрелка, указывающая на другую модель работы. Стрелка вызова рисуется как исходящая из нижней грани работы. Стрелка вызова используется для указания того, что некоторая работа выполняется за пределами моделируемой системы. В VPwin стрелки вызова используются в механизме слияния и разделения моделей.

Граничные стрелки.

Стрелки на контекстной диаграмме служат для описания взаимодействия системы с окружающим миром. Они могут начинаться у границы диаграммы и заканчиваться у работы, или наоборот. Такие стрелки называются граничными.

Для внесения граничной стрелки входа следует:

- щелкнуть по кнопке с символом стрелки ,
- в палитре инструментов перенести курсор к левой стороне экрана, пока не появится начальная штриховая полоска;
- щелкнуть один раз по полоске (откуда выходит стрелка) и еще раз в левой части работы со стороны входа (где заканчивается стрелка);
- вернуться в палитру инструментов и выбрать опцию редактирования стрелки 
- щелкнуть правой кнопкой мыши на линии стрелки, во всплывающем меню выбрать Name и добавить имя стрелки в закладке Name диалога IDEF0 Arrow Properties.

Стрелки управления, входа, механизма и выхода изображаются аналогично. Имена вновь внесенных стрелок автоматически заносятся в словарь Arrow Dictionary.

ICOM-коды.

Диаграмма декомпозиции предназначена для детализации работы. В отличие от моделей, отображающих структуру организации, работа на диаграмме верхнего уровня в IDEF0 — это не элемент управления

нижестоящими работами. Работы нижнего уровня — это то же самое, что работы верхнего уровня, но в более детальном изложении. Как следствие этого границы работы верхнего уровня — это то же самое, что границы диаграммы декомпозиции.

ICOM (аббревиатура от Input, Control, Output и Mechanism) — коды, предназначенные для идентификации граничных стрелок. Код ICOM содержит префикс, соответствующий типу стрелки (I, C, O или M), и порядковый номер. VPwin вносит ICOM-коды автоматически. Для отображения ICOM-кодов следует включить опцию ICOM codes на закладке Display диалога Model Properties (меню Model/Model Properties).

Словарь стрелок редактируется при помощи специального редактора Arrow Dictionary Editor, в котором определяется стрелка и вносится относящийся к ней комментарий. Содержимое словаря стрелок можно распечатать в виде отчета (меню Tools/ Report /Arrow Report...) и получить толковый словарь терминов предметной области, использующихся в модели.

Несвязанные граничные стрелки (unconnected border arrow).

При декомпозиции работы входящие в нее и исходящие из нее стрелки (кроме стрелки вызова) автоматически появляются на диаграмме декомпозиции (миграция стрелок), но при этом не касаются работ. Такие стрелки называются несвязанными и воспринимаются в VPwin как синтаксическая ошибка.

IDEF3

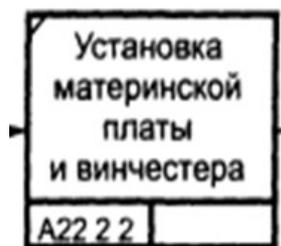
IDEF3 называют также **workflow diagramming** - методология моделирования, использующая графическое описание информационных потоков, взаимоотношений между процессами обработки информации и объектов, являющихся частью этих процессов.

IDEF3 — это метод, имеющий основной целью дать возможность аналитикам описать ситуацию, когда процессы выполняются в определенной последовательности, а также описать объекты, участвующие совместно в одном процессе.



Единицы работы

Единицы работы — Unit of Work (UOW) — также называемые работами (activity), являются центральными компонентами модели. В IDEF3 работы изображаются прямоугольниками с прямыми углами и имеют имя, выраженное отглагольным существительным, обозначающим процесс действия, одиночным или в составе фразы, и номер (идентификатор); другое имя существительное в составе той же фразы обычно отображает основной выход (результат) работы (например, "Изготовление изделия"). Часто имя существительное в имени работы меняется в процессе моделирования, поскольку модель может уточняться и редактироваться. Идентификатор работы присваивается при создании и не меняется никогда. Даже если работа будет удалена, ее идентификатор не будет вновь использоваться для других работ. Обычно номер работы состоит из номера родительской работы, точки и порядкового номера на текущей диаграмме.



Связи

Связи показывают взаимоотношения работ. Все связи в IDEF3 однонаправлены и могут быть направлены куда угодно, но обычно диаграммы IDEF3 стараются построить так, чтобы связи были направлены слева направо. В

IDEF3 различают три типа стрелок, изображающих связи, стиль которых устанавливается через меню Edit/Arrow Style:

Старшая (Precedence) – временное предшествование



→ сплошная линия, связывающая единицы работ (UOW). Рисуеться слева направо или сверху вниз. Показывает, что работа-источник должна закончиться прежде, чем работа-цель начнется.

Отношения (Relational Link) – нечеткое отношение



⋯→ пунктирная линия, используемая для изображения связей между единицами работ (UOW) а также между единицами работ и объектами ссылок.

Потоки объектов (Object Flow) – объектный поток



⇒ стрелка с двумя наконечниками, применяется для описания того факта, что объект используется в двух или более единицах работы, например, когда объект порождается в одной работе и используется в другой.

Старшая связь показывает, что исходное действие должно полностью завершиться, прежде чем начнется выполнение конечного действия.

Часто результатом исходного действия становится объект, необходимый для запуска конечного действия. В этом случае стрелку, обозначающую объект, изображают с двойным наконечником. Имя стрелки должно ясно идентифицировать отображаемый объект. Поток объектов имеет ту же семантику, что и старшая стрелка.

Отношение показывает, что стрелка является альтернативой старшей стрелке или потоку объектов в смысле задания последовательности выполнения работ — предшествующее действие не обязательно должна закончиться, прежде чем конечное действие начнется. Более того, конечное действие может закончиться прежде, чем закончится предшествующее действие.

Окончание одной работы может служить сигналом к началу нескольких работ, или же одна работа для своего запуска может ожидать окончания нескольких работ.

Соединения (перекрестки)

Для отображения логики взаимодействия стрелок при слиянии и разветвлении или для отображения множества событий, которые могут или должны быть завершены перед началом следующей работы, используются Соединения (Junction) – или перекрестки.

Различают соединения для слияния (Fan-in Junction) и разветвления стрелок (Fan-out Junction). Соединение не может использоваться одновременно для слияния и для разветвления.

Все соединения на диаграмме нумеруются, каждый номер имеет префикс J. В отличие от IDEF0 в IDEF3 стрелки могут сливаться и разветвляться только через перекрестки.

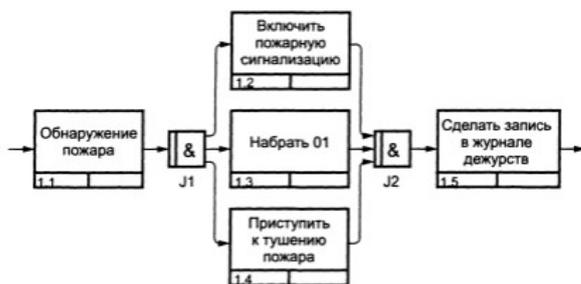
Для внесения соединения служит кнопка — (добавить в диаграмму соединение — Junction) в палитре инструментов. В диалоге Select Junction Type необходимо указать тип соединения. Смысл каждого типа приведен в таблице.

Графическое обозначение	Название	Вид	Правила инициации
&	Соединение "И"	Разворачивающее	Каждое конечное действие обязательно иницируется
		Сворачивающее	Каждое исходное действие обязательно должно завершиться
X	Соединение "Эксклюзивное ИЛИ"	Разворачивающее	Одно и только одно конечное действие иницируется
		Сворачивающее	Одно и только одно исходное действие должно завершиться
O	Соединение "ИЛИ"	Разворачивающее	Одно (или более) конечное действие иницируется
		Сворачивающее	Одно (или более) исходное действие должно завершиться

"И"-соединения (Перекресток И).

Соединения этого типа иницируют выполнение всех своих конечных действий. Все действия, присоединенные к сворачивающему "И"-соединению, должны завершиться, прежде чем может начать выполняться следующее действие. На рисунке после обнаружения пожара иницируются включение пожарной сигнализации, вызов пожарной охраны и начинается тушение

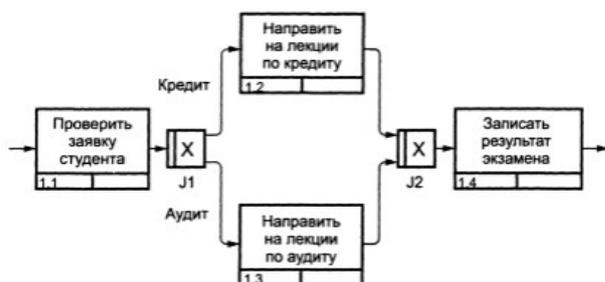
пожара. Запись в журнал производится только тогда, когда все три перечисленных действия завершены.



Соединение "Эксклюзивное ИЛИ".

Вне зависимости от количества действий, прицепленных к сворачивающему или разворачивающему соединению "Эксклюзивное ИЛИ", инициировано будет только одно из них, и поэтому только одно из них будет завершено перед тем, как любое действие, следующее за сворачивающим соединением "Эксклюзивное ИЛИ", сможет начаться.

Если правила активации соединения известны, они обязательно должны быть документированы либо в его описании, либо пометкой стрелок, исходящих из разворачивающего соединения, как показано на рисунке. На рисунке соединение "Эксклюзивное ИЛИ" используется для отображения того факта, что студент не может одновременно быть направлен на лекции по двум разным курсам.



Соединение "ИЛИ"

Соединения этого типа предназначены для описания ситуаций, которые не могут быть описаны двумя предыдущими типами соединений. На рисунке соединение J2 может активировать проверку данных чека и (или) проверку суммы наличных. Проверка чека инициируется, если покупатель желает расплатиться чеком, проверка суммы наличных — при оплате наличными. И то,

и другое действие инициируется при частичной оплате чеком и частичной — наличными.



Синхронные и асинхронные соединения

В рассмотренных примерах связей "И" и "ИЛИ" мы не затрагивали отношений между началом и окончанием действий, инициируемых разворачивающими соединениями. Все действия в этих примерах выполнялись асинхронно, т.е. они не должны были начинать выполняться одновременно. Однако есть случаи, когда время начала или окончания параллельно выполняемых действий должно быть одинаковым, т.е. действия должны выполняться синхронно. Для моделирования такого поведения системы используются синхронные соединения.

Графическое обозначение	Тип	Вид	Правила инициации
⊗	И	Разворачивающее	Все действия начнутся одновременно
		Сворачивающее	Все действия закончатся одновременно
⊙	ИЛИ	Разворачивающее	Может быть, несколько действий начнутся одновременно
		Сворачивающее	Может быть, несколько действий закончатся одновременно
⊗	Эксклюзивное ИЛИ	Разворачивающее	Одновременное начало действий невозможно
		Сворачивающее	Одновременное окончание действий невозможно

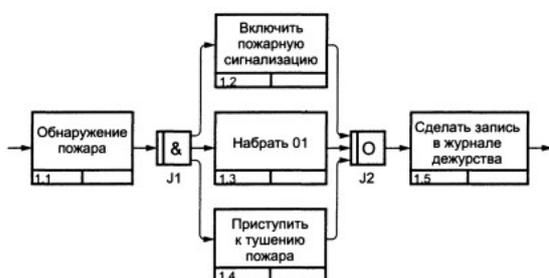
Синхронное соединение обозначается двумя вертикальными линиями внутри обозначающего его прямоугольника в отличие от одной вертикальной линии в асинхронном соединении. Во многих спортивных состязаниях выстрел стартового пистолета, запуск секундомера и начало состязания должны произойти одновременно. В противном случае состязание будет нечестным. Рисунок иллюстрирует модель этого примера, построенную с использованием синхронного соединения.



Заметим, что синхронное разворачивающее соединение не обязательно должно иметь парное себе сворачивающее соединение. Действительно, начинающиеся одновременно действия вовсе не обязаны оканчиваться одновременно, как это видно из примера с состязаниями. Также возможны ситуации синхронного окончания асинхронно начавшихся действий.

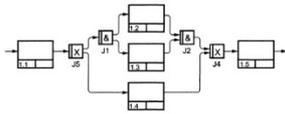
Парность соединений

Все соединения на диаграммах должны быть парными, из чего следует, что любое разворачивающее соединение имеет парное себе сворачивающее. Однако типы соединений вовсе не обязательно должны совпадать. Например, разворачивающее "И"-соединение имеет парное сворачивающее "ИЛИ"-соединение. Интерпретация соединения J1 - после обнаружения пожара инициируются включение пожарной сигнализации, вызов пожарной охраны и начинается тушение пожара. Соединение J2 интерпретируется следующим образом: после включения пожарной сигнализации и (или) вызова пожарных и (или) начала тушения производится запись в журнал.



Комбинации соединений

Соединения могут комбинироваться для создания более сложных правил ветвления.



Комбинации соединений следует использовать с осторожностью, поскольку перегруженные ветвлением диаграммы могут оказаться сложными для восприятия

Указатели

Указатели (объект ссылки) — это специальные символы, которые ссылаются на другие разделы описания процесса. Они выносятся на диаграмму для привлечения внимания читателя к каким-либо важным аспектам модели.

Для внесения указателя служит кнопка — (добавить в диаграмму объект ссылки — Referent) в палитре инструментов, изображается в виде прямоугольника, похожего на прямоугольник работы

Имя указателя задается в диалоге Referent (пункт Name контекстного меню), в качестве имени можно использовать имя какой-либо стрелки с других диаграмм или имя сущности из модели данных. Указатели должны быть связаны с единицами работ или соединениями пунктирными линиями.

Таблица 8.2. Типы объектов ссылок

Тип объекта ссылки	Цель описания
ОБЪЕКТ - OBJECT	Описывает участие важного объекта в работе
ССЫЛКА - GOTO	Для реализации цикличности выполненных действий. GOTO может ссылаться на соединение
ЕДИНИЦА ДЕЙСТВИЯ - UOB (Unit of behaviour)	Применяется, когда необходимо подчеркнуть множественное использование какой-либо работы, но без цикла. Например, работа "Контроль качества" может быть использована в <i>процессе</i> "Изготовление изделия" несколько раз, после каждой единичной операции. Обычно этот тип ссылки не используется для моделирования автоматически запускающихся работ
ЗАМЕТКА - NOTE	Используется для документирования важной информации,

	относящейся к каким-либо графическим объектам на диаграмме. NOTE является альтернативой внесению текстового объекта в диаграмму
УТОЧНЕНИЕ - ELAB (Elaboration)	Используется для усовершенствования графиков или их более детального описания. Обычно употребляется для детального описания разветвления и слияния стрелок на <i>соединениях</i>

DFD - Диаграммы потоков данных

Диаграммы потоков данных (Data Flow Diagramming) моделируют систему как набор работ, соединенных друг с другом стрелками, которые показывают, как каждая работа преобразует свои входные данные в выходные, и выявляет отношения между этими работами.

Любая DFD-диаграмма может содержать работы (процессы), хранилища данных (накопители данных), внешние сущности, стрелки (потоки данных).

Работы. (Процессы) моделируют некоторую функцию, которая преобразует сырье в какую-либо продукцию. Смысл их совпадает со смыслом работ IDEF0 и IDEF3. Так же как работы IDEF3, они имеют входы и выходы, но не поддерживают управления и механизмы, как IDEF0. Все стороны работы равнозначны. В каждую работу может входить и выходить по несколько стрелок.

Внешние сущности - материальный объект (заказчики, персонал, поставщики, клиенты, склад) или процессы, внешние по отношению к системе, т.е. находящиеся за пределами границ анализируемой системы. Одна внешняя сущность может одновременно предоставлять входы (поставщик) и принимать выходы (получатель).

Одна внешняя сущность может быть использована многократно на одной или нескольких диаграммах. Обычно такой прием используют, чтобы не рисовать слишком длинных и запутанных стрелок.

Потоки работ изображаются стрелками и описывают движение объектов из одной части системы в другую, т.е. как объекты (включая данные) двигаются и изменяются от одной работы к другой. (отсюда следует, что диаграмма DFD не может иметь граничных стрелок). Поскольку в DFD каждая сторона работы не имеет четкого назначения, как в IDEF0, стрелки могут подходить и выходить из любой грани прямоугольника работы. В DFD также применяются двунаправленные стрелки для описания диалогов типа "команда-ответ" между работами, между работой и внешней сущностью и между внешними сущностями.

В отличие от стрелок, описывающих объекты в движении, хранилища данных изображают объекты в покое

Хранилище данных - это абстрактное устройство для хранения информации, которую можно в любой момент поместить в накопитель и через некоторое время извлечь, причем способы помещения и извлечения могут быть любыми.

Оно в общем случае является прообразом будущей базы данных, и описание хранящихся в нем данных должно соответствовать информационной модели (Entity-Relationship Diagram). В материальных системах хранилища данных изображаются там, где объекты ожидают обработки, например в очереди. В системах обработки информации хранилища данных являются механизмом, который позволяет сохранить данные для последующих процессов.

Представление потоков работ совместно с хранилищами данных и внешними сущностями делает модели DFD более похожими на физические характеристики системы — движение объектов, хранение объектов, поставка и распространение объектов.

Построение DFD-диаграмм ассоциируется с разработкой программного обеспечения, поскольку DFD изначально была разработана именно для этого.

Для описания диаграмм DFD используются две нотации — Йордана-ДеМарко (Yourdon-DeMarco) и Гейна-Сарсона (Gane-Sarson), отличающиеся синтаксисом, обе предложенные в 1979 году.

Нотация Йордана-ДеМарко:

Процессы изображаются кружками,
внешние сущности — прямоугольниками,
хранилища данных — двумя горизонтальными параллельными линиями.

Нотация Гейна-Сарсона:

процессы — прямоугольники со скругленными углами,
внешние сущности — прямоугольники с тенью,
хранилища данных — вытянутые горизонтально прямоугольники без
правого ребра.

В ВРwin для построения диаграмм потоков данных используется нотация Гейна-Сарсона.

Для того чтобы дополнить модель IDEF0 диаграммой DFD, нужно в процессе декомпозиции в диалоге Activity Box Count "кликнуть" по радио-кнопке DFD.

Синтаксис и семантика DFD

В отличие от IDEF0, где система рассматривается как взаимосвязанные работы, DFD рассматривает систему как совокупность объектов. Названия объектов в DFD обычно имена существительные.

Контекстная диаграмма потоков данных должна содержать нулевую работу с именем, отражающим деятельность организации, а также несколько внешних сущностей, соединенные с нулевой работой потоками данных. Потоки данных соответствуют документам, запросам или сообщениям, которыми внешние сущности обмениваются с организацией.

Например, контекстная диаграммы системы, автоматизирующей деятельность отдела по работе с банковскими пластиковыми картами.



Нумерация работ

В DFD номер каждой работы может включать префикс, номер родительской работы и номер объекта. Номер объекта — это уникальный номер работы на диаграмме. Например, работа может иметь номер A.12.4. Уникальный номер имеют хранилища данных и внешние сущности независимо от их расположения на диаграмме. Каждое хранилище данных имеет префикс D и уникальный номер, например D5. Каждая внешняя сущность имеет префикс E и уникальный номер, например E5.

Декомпозиция работы IDEF0 в диаграмму DFD.

1 - удалить все граничные стрелки на диаграмме DFD или заменить их на соответствующие внешние сущности и хранилища данных. Стрелки механизмов, стрелки управления как правило на дочерней диаграмме задействованы не будут.

2 - создать внутренние стрелки, начинающиеся с внешних сущностей вместо граничных стрелок; Входом работ и выходом будут некоторые внешние сущности или другие работы

3 - определить, какая информация необходима для каждой работы, т.е. необходимо разместить на диаграмме хранилища данных

4 - стрелки на диаграмме IDEF0 затоннелировать

Методические рекомендации по подготовке сообщения и презентации

Задание:

Подготовить презентацию, на которой представить разработанную в предыдущем самостоятельном практическом задании по выбранной предметной области диаграмму декомпозиции первого уровня модели IDEF0. В сообщении обосновать, почему модель была декомпозирована именно так.

Презентацию подготовить в Power Point. Задание выполняется в микрогруппе по 2-3 человека (как в предыдущем задании).

2.2 РАЗДЕЛ 3. МЕТОД СУЩНОСТЬ-СВЯЗЬ (ER)

Тема 3.1. Метод Сущность-связь (ER)

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников.

Ключевым для данной темы являются понятия Entity-Relationship Diagram, сущность, связь, реляционная база данных, отношение, таблица, атрибут, запись.

Студенты должны знать процесс нормализации базы данных, технологию приведения отношения в третью нормальную форму, избавления от избыточности и обеспечения целостности базы данных.

Методические рекомендации по подготовке сообщения и презентации

Задание:

1 Используя предложенную литературу, а также Интернет-источники, найти различные нотации Entity-Relationship Diagram.

2 Подготовить сообщение и презентацию Power Point, на которой представить выбранную нотацию (по согласованию с преподавателем). Для нотации представить следующее описание:

- Разработчик и год появления
- Используемые графические элементы
- С какой целью данная нотация была разработана, где ее применяют
- Пример модели в данной нотации (в виде рисунка)

3 Задание выполняется в микрогруппе по 2-3 человека.

2.3 РАЗДЕЛ 4. ВИЗУАЛЬНОЕ МОДЕЛИРОВАНИЕ НА ЯЗЫКЕ UML

Тема 4.1. Визуальное моделирование на языке UML

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников и выполнение самостоятельных практических заданий.

Ключевым для данной темы являются понятия UML, сущности UML, диаграммы UML, диаграммы UML.

Студенты должны знать технологию построения диаграмм UML: диаграмма вариантов использования, диаграмма классов, диаграмма последовательности, диаграмма состояний.

Самостоятельное задание 1: Создание диаграммы вариантов использования (по вариантам).

Спроектировать диаграмму вариантов использования одного из предложенных вариантов. Вариант назначает преподаватель. Задание выполняется индивидуально.

Вариант 1 Система табельного учета

Задачи контроля рабочего времени, выполняемые операционистом отдела кадров, включают сбор отпечатанных табелей, проверку полноты и

правильности заполнения таблиц и ввод данных табельных карточек в действующую систему расчета заработной платы, чтобы подготовить печать платежной ведомости.

Кроме того, сотрудник отдела кадров должен поддерживать коды зарплаты (обычное время, сверхурочное время, отпуск по болезни и т.д.) и распределять список кодов среди служащих компании при каждом его изменении.

Эта работа в свою очередь влияет на способность главных специалистов отдела выполнять другие обязанности, в том числе готовить отчеты по зарплате, начислять пособия и выполнять функции найма, продвижения и увольнения.

Успешное решение проблемы должно сократить время, которое отдел кадров тратит на задачи, связанные с табельным учетом, на 60% и более.

Вариант 2 Каталог Интернет-ресурсов

В каталоге хранится следующая информация о Интернет-ресурсах: название ресурса, уникальный локатор ресурса (URL), раздел каталога, в котором содержится ресурс, список ключевых слов, краткое описание, дата последнего обновления, контактная информация.

Доступ пользователей к каталогу осуществляется при помощи браузера. Пользователи каталога могут добавлять новые ресурсы, информация о которых не была внесена ранее. Добавлять информация может только зарегистрированный пользователь. Учет пользователей ведет модератор каталога.

Полный список ресурсов доступен для просмотра как зарегистрированным пользователям, так и незарегистрированным гостям каталога.

Пользователям и гостям каталога должны быть предоставлены возможности по поиску ресурсов. Должна быть возможность выдавать результаты поиска. Пользователь и гость может отсортировать список ресурсов

по релевантности (соответствию ключевым словам из запроса) или по дате обновления.

Поскольку содержание ресурсов Интернет со временем изменяется необходимо следить за датой последнего обновления, периодически опрашивая Web-сайты, URL которых хранятся в каталоге. Этим занимается модератор каталога. При необходимости модератор может изменять данные ресурса.

Система учета по требованию модератора формирует и выдает на печать список необновляемых ресурсов, которые модератор может удалить.

Вариант 3 Электронная проходная

Для грамотного и оперативного учета рабочего времени на предприятии было решено разработать и внедрить систему электронной проходной. Для осуществления данного проекта каждый сотрудник предприятия получает электронный пропуск, на котором указывается необходимая информация о сотруднике. Электронный пропуск программируется специальным устройством – программатором.

Система электронной проходной должна идентифицировать сотрудника, выводить на монитор электронной проходной фотографию сотрудника, получать сведения о приходе и уходе сотрудника и сохранять их в базе данных. Время прихода и ухода сотрудника на работу система электронной проходной должна пересылать в систему расчета заработной платы.

Система электронной проходной включает также функции отдела кадров. Сотрудники отдела кадров ведут учет сотрудников (оформляют документы приема на работу, отслеживают повышение квалификации работника и оформляют увольнение).

Методические рекомендации по выполнению задания:

Дважды щелкните на Главной диаграмме Вариантов Использования (Main) в браузере, чтобы открыть ее.

С помощью кнопки Use Case (Вариант Использования) панели инструментов поместите на диаграмму новый вариант использования.

Повторите этапы, чтобы поместить на диаграмму остальные варианты использования.

С помощью кнопки Actor (Действующее лицо) панели инструментов поместите на диаграмму новое действующее лицо. Повторите шаги, поместив на диаграмму остальных действующих лиц.

Добавить ассоциации.

С помощью кнопки Unidirectional Association (Однонаправленная ассоциация) панели инструментов нарисуйте ассоциацию между действующим лицом и вариантом использования.

Повторите этот этап, чтобы поместить на диаграмму остальные ассоциации.

Добавить описания к вариантам использования

Выделите в браузере вариант использования. В окне документации введите описание к этому варианту использования.

Добавить описания к действующему лицу

Выделите в браузере действующее лицо. В окне документации введите для этого действующего лица описание.

Критерии оценки практического задания:

- Полнота диаграммы;
- Обоснованность решения;
- Стиль проектирования;

Оценка «Отлично» ставится за полноту диаграммы, грамотную структуру и стиль оформления.

Оценка «Хорошо» ставится за полноту диаграммы, замечания по структуре и стилю оформления.

Оценка «Удовлетворительно» ставится за неполноту диаграммы, замечания по структуре и стилю оформления.

Самостоятельное задание 2. Создание диаграммы классов.

Спроектировать диаграмму классов варианта, полученного в самостоятельном задании 1. Задание выполняется индивидуально.

Методические рекомендации по выполнению задания:

Спроектировать классы для каждого варианта использования, для которых сформированы сценарии. Определить первоначальный набор атрибутов и ассоциаций классов (диаграмма классов для каждого созданного сценария варианта использования уровня предварительного проектирования). Для каждого класса указать стереотип. На диаграмме должны присутствовать классы со стереотипами: граничный, сущность, управляющий. Поместить созданные диаграммы в виде картинок в текстовый документ с описанием проекта.

Спроектировать пакеты (если это необходимо). Пакеты определить самостоятельно, исходя из предметной области и выбранной технологии проектирования. Разместить классы по пакетам. В документ проекта поместить в виде картинки представление логического браузера с размещенными по пакетам классами.

Спроектировать (любыми средствами) модель предметной области уровня предварительного проектирования.

Уточнить операции классов (Определить полную сигнатуру) и уточнить атрибуты классов (тип атрибута).

Для каждого спроектированного класса в текстовом документе проекта выполнить описание, в котором представить в виде таблицы: название операции/атрибута, сигнатура (описание или тип), назначение.

Спроектировать общую диаграмму классов. Уточнить связи между классами на диаграмме.

Поместить модель в отчет в виде картинки.

Критерии оценки практического задания:

- Полнота диаграммы;
- Обоснованность решения;

- Стиль проектирования;

Оценка «Отлично» ставится за полноту диаграммы, грамотную структуру и стиль оформления.

Оценка «Хорошо» ставится за полноту диаграммы, замечания по структуре и стилю оформления.

Оценка «Удовлетворительно» ставится за неполноту диаграммы, замечания по структуре и стилю оформления.

Методические рекомендации по подготовке сообщения и презентации

Задание:

Используя конспекты лекций, предложенную литературу, а также Интернет-источники, подготовить сообщение и презентацию Power Point по анализу одной из сущностей языка UML (по согласованию с преподавателем). Для сущностей представить следующее описание:

- Наименование и графическое изображение
- Назначение
- В каких диаграммах UML используется
- Пример диаграмм (в виде рисунка)

Задание выполняется в микрогруппе по 2-3 человека.

Рассматриваемые сущности:

- Актер
- Вариант использования
- Класс
- Объект
- Состояние
- Узел
- Компонент
- Взаимодействие

Тема 5.1. Жизненный цикл ПО**Методические рекомендации по подготовке к практической работе**

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников.

Ключевым для данной темы являются понятия: жизненный цикл программного обеспечения, процессы жизненного цикла программного обеспечения, модели жизненного цикла программного обеспечения.

Студенты должны знать ГОСТ 12207-2010 и основные понятия, прописанные в данном ГОСТе.

Методические рекомендации по подготовке сообщения и презентации**Задание:**

Используя конспекты лекций, предложенную литературу, а также Интернет-источники, подготовить сообщение и презентацию Power Point по анализу одной из моделей жизненного цикла программного обеспечения (по согласованию с преподавателем). Модели анализировать по следующим категориям:

- Наименование
- Принцип построения модели
- Достоинства модели
- Недостатки модели

Задание выполняется в микрогруппе по 2-3 человека.

Рассматриваемые модели:

- Каскадная модель
- Спиральная модель

- Итерационная модель
- Инкрементная модель
- Модель быстрого прототипирования
- Модель RUP
- Объектная модель

Тема 5.2. Технологии создания ПО

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников.

Ключевым для данной темы являются понятия технологии проектирования программного обеспечения, канонического и типового проектирования программного обеспечения.

Студенты должны знать технологии RUP, RAD, eXP, ICONIX.

2.5 РАЗДЕЛ 6. ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Тема 6.1. Моделирование прецедентов

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников и выполнение самостоятельного практического задания.

Ключевым для данной темы являются понятия технологии объектно-ориентированного проектирования ICONIX, этап анализа технологии ICONIX.

Самостоятельное практическое задание:

Для выбранной предметной области спроектировать модель вариантов использования.

Цель выполнения задания - закрепление практических навыков проектирования модели вариантов использования

Задание:

С целью проведения этапа анализа технологии ICONIX создайте модель вариантов использования проекта информационной системы выбранного предприятия

Задание выполняется в микрогруппе по 2-3 человека. Каждая микрогруппа выбирает один из вариантов предложенных предметных областей. Выбор согласовывается с преподавателем.

Сохранить созданные модели в виде картинок в документе Word. Сформированный документ представить преподавателю для проверки.

Варианты предметных областей:

1. Система автоматизации гостиницы
2. Система автоматизации ресторана
3. CRM-система
4. Библиотечная система
5. Система автоматизации автобусного парка
6. Система автоматизации парковки
7. Система автоматизации проката автомобиля
8. Система кредитования банка
9. Система автоматизации турбазы
10. Система автоматизации салона красоты
11. Система автоматизации фитнес-центра

Методические рекомендации по выполнению самостоятельного практического задания:

Дважды щелкните на Главной диаграмме Вариантов Использования (Main) в броузере, чтобы открыть ее.

С помощью кнопки Use Case (Вариант Использования) панели инструментов поместите на диаграмму новый вариант использования. Повторите этапы, чтобы поместить на диаграмму остальные варианты использования.

С помощью кнопки Actor (Действующее лицо) панели инструментов поместите на диаграмму новое действующее лицо. Повторите шаги, поместив на диаграмму остальных действующих лиц.

Добавить ассоциации.

С помощью кнопки Unidirectional Association (Однонаправленная ассоциация) панели инструментов нарисуйте ассоциацию между действующим лицом и вариантом использования.

Повторите этот этап, чтобы поместить на диаграмму остальные ассоциации.

Добавить описания к вариантам использования

Выделите в браузере вариант использования. В окне документации введите описание к этому варианту использования.

Добавить описания к действующему лицу

Выделите в браузере действующее лицо. В окне документации введите для этого действующего лица описание.

Критерии оценки практического задания:

- Полнота решения диаграммы;
- Стиль оформления
- Грамотность изложения;

Оценка «Отлично» ставится за грамотно спроектированную диаграмму, стильно оформленный прототип интерфейса, грамотную структуру и стиль оформления задания.

Оценка «Хорошо» ставится за спроектированную диаграмму с ошибками, негармонично оформленный прототип интерфейса, грамотную структуру и стиль оформления задания с замечаниями.

Оценка «Удовлетворительно» ставится за спроектированную диаграмму с грубыми ошибками, негармонично оформленный прототип интерфейса, грамотную структуру и стиль оформления задания с существенными замечаниями.

Методические рекомендации по подготовке сообщения и презентации

Задание:

Используя конспект лекций, предложенную литературу, а также Интернет-источники, подготовить сообщение и презентацию Power Point, на которой представить последовательность проектирования программного обеспечения по технологии ICONIX (этапы технологии ICONIX) (по согласованию с преподавателем).

Задание выполняется в микрогруппе по 2-3 человека.

Рассматриваемые этапы:

- Архитектурный анализ
- Проектирование вариантов использования
- Спецификация вариантов использования
- Анализ вариантов использования
- Проектирование диаграммы последовательности
- Проектирование диаграммы классов
- Проектирование базы данных
- Этап реализации

Тема 6.2. Проектирование пользовательского интерфейса

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников.

Ключевым для данной темы являются понятия технологии объектно-ориентированного проектирования ICONIX, этап предварительного проектирования технологии ICONIX. Принципы проектирования пользовательского интерфейса.

Тема 6.3. Проектирование баз данных информационных систем

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников.

Ключевым для данной темы являются понятия технологии объектно-ориентированного проектирования ICONIX, этап детального проектирования технологии ICONIX. Принципы проектирования баз данных.

Тема 6.4. Проектирование классов

Методические рекомендации по подготовке к практической работе

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников и выполнение самостоятельного практического задания.

Ключевым для данной темы являются понятия технологии объектно-ориентированного проектирования ICONIX, этап детального проектирования технологии ICONIX. Стереотипы классов.

Самостоятельное практическое задание:

Для выбранной предметной области спроектировать диаграмму классов.

Цель выполнения задания - закрепление практических навыков проектирования диаграммы классов.

Задание:

С целью проведения этапа детального проектирования технологии ICONIX создайте диаграмму классов проекта информационной системы выбранного предприятия

Задание выполняется в микрогруппе по 2-3 человека. Каждая микрогруппа выбирает один из вариантов предложенных предметных областей. Выбор согласовывается с преподавателем.

Сохранить созданные модели в виде картинок в документе Word. Сформированный документ представить преподавателю для проверки.

Варианты предметных областей:

1. Система автоматизации гостиницы
2. Система автоматизации ресторана
3. CRM-система
4. Библиотечная система
5. Система автоматизации автобусного парка
6. Система автоматизации парковки
7. Система автоматизации проката автомобиля
8. Система кредитования банка
9. Система автоматизации турбазы
10. Система автоматизации салона красоты
11. Система автоматизации фитнес-центра

Методические рекомендации по выполнению самостоятельного практического задания:

Спроектировать классы для каждого варианта использования, для которых сформированы сценарии. Определить первоначальный набор атрибутов и ассоциаций классов (диаграмма классов для каждого созданного сценария варианта использования уровня предварительного проектирования).

Для каждого класса указать стереотип. На диаграмме должны присутствовать классы со стереотипами: граничный, сущность, управляющий. Поместить созданные диаграммы в виде картинок в текстовый документ с описанием проекта.

Спроектировать пакеты (если это необходимо). Пакеты определить самостоятельно, исходя из предметной области и выбранной технологии проектирования. Разместить классы по пакетам. В документ проекта поместить в виде картинки представление логического браузера с размещенными по пакетам классами.

Спроектировать (любыми средствами) модель предметной области уровня предварительного проектирования.

Уточнить операции классов (Определить полную сигнатуру) и уточнить атрибуты классов (тип атрибута).

Для каждого спроектированного класса в текстовом документе проекта выполнить описание, в котором представить в виде таблицы: название операции/атрибута, сигнатура (описание или тип), назначение.

Спроектировать общую диаграмму классов. Уточнить связи между классами на диаграмме.

Поместить модель в отчет в виде картинки.

Критерии оценки практического задания:

- Полнота решения диаграммы;
- Стиль оформления
- Грамотность изложения;

Оценка «Отлично» ставится за грамотно спроектированную диаграмму, стильно оформленный прототип интерфейса, грамотную структуру и стиль оформления задания.

Оценка «Хорошо» ставится за спроектированную диаграмму с ошибками, негармонично оформленный прототип интерфейса, грамотную структуру и стиль оформление задания с замечаниями.

Оценка «Удовлетворительно» ставится за спроектированную диаграмму с грубыми ошибками, негармонично оформленный прототип интерфейса, грамотную структуру и стиль оформление задания с существенными замечаниями.