

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»**

На правах рукописи

Канев Дмитрий Сергеевич

**РАЗРАБОТКА МОДЕЛЕЙ, МЕТОДОВ И СРЕДСТВ ФОРМИРОВАНИЯ
ПРОФИЛЯ КОМПЕТЕНЦИЙ ПРОЕКТИРОВЩИКА В ПРОЦЕССАХ
АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ
МАШИНОСТРОИТЕЛЬНЫХ ОБЪЕКТОВ (НА ПРИМЕРЕ САПР
КОМПАС)**

Специальность: 05.13.12 – Системы автоматизации
проектирования (промышленность)

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель – доктор технических наук, профессор,
Афанасьев Александр Николаевич

Ульяновск – 2016

Оглавление

Принятые сокращения и обозначения.....	5
Введение.....	6
Глава 1. Компьютерные системы формирования профиля проектировщика в процессах автоматизированного проектирования машиностроительных объектов	11
1.1. Профиль компетенций проектировщика машиностроительных объектов в условиях применения САПР.....	11
1.2. Методы и средства адаптивного управления процессом обучения в автоматизированных обучающих системах для формирования профиля проектировщика	14
1.3. Анализ систем рекомендаций для проектировщиков	29
1.4. Основные подходы к проектированию виртуальных компонентов автоматизированных обучающих систем.....	37
1.5. Разработка обобщенной схемы формирования профиля компетенций проектировщика в процессах автоматизированного проектирования	40
1.6. Постановка задачи.....	42
1.7. Выводы.....	44
Глава 2. Разработка математического обеспечения системы формирования профиля компетенций проектировщика.....	48
2.1. Общая организация системы формирования профиля компетенций проектировщика	48
2.2. Разработка процесса формирования профиля компетенций: режим сценария, режим поддержки проектировщика	50
2.3. Разработка моделей предметной области, профиля проектировщика и контроля	54

2.4. Разработка метода формирования персонифицированного сценария обучения.....	68
2.5. Выводы и рекомендации	73
Глава 3. Разработка системы рекомендаций и виртуальной компоненты для формирования профиля компетенций проектировщика.....	75
3.1. Формальное описание задачи	75
3.2. Формирование списка рекомендаций	77
3.3. Классификация рекомендаций.....	88
3.4. Разработка метода формирования рекомендаций	90
3.5. Разработка модели виртуального компонента автоматизированной обучающей системы.....	96
3.6. Разработка метода автоматической диагностики ошибок обучаемого при работе с виртуальными компонентами автоматизированной обучающей системы	99
3.7. Выводы и рекомендации	102
Глава 4. Реализация компьютерной системы формирования профиля компетенций проектировщика	104
4.1. Разработка компонентной архитектуры системы формирования профиля компетенций проектировщика.....	104
4.2. Реализация графического редактора модели предметной области	108
4.3. Реализация метода формирования персонифицированного сценария обучения.....	113
4.4. Реализация системы рекомендаций для проектировщика.....	116
4.5. Оценка метода формирования персонифицированного сценария обучения	123

4.6. Оценка повышения эффективности деятельности проектировщика при использовании системы рекомендаций	129
4.7. Реализация виртуальных компонент для системы формирования профиля компетенций проектировщика.....	133
4.8. Реализации ядра мобильной системы обучения, моделей предметной области, обучаемого, тестирования для платформы Android.....	137
4.9. Выводы и рекомендации	148
Заключение	150
Список литературы	151
Приложение 1. Список знаний, умений и навыков для компетенции «владеет навыками твердотельного моделирования»	168
Приложение 2. Перечень рекомендаций.....	173
Приложение 3. Описание контроллеров для виртуальных компонент системы формирования профиля компетенций проектировщика.....	176
Приложение 4. Анализ сборки «Насос»	179
Приложение 5. Акт о внедрении.....	186

Принятые сокращения и обозначения

API – Application Programming Interface

CAD – Computer-aided design

DOM – Document Object Model

REST – Representational state transfer

RMI – Remote Method Invocation

SOAP – Simple Object Access Protocol

TF-IDF – Term Frequency — Inverse Document Frequency

UML – Unified Modeling Language

WSDL – Web Services Description Language

XML – eXtensible Markup Language

АОС – автоматизированная обучающая система

БД – база данных

ПК – профессиональные компетенции

ПрО – предметная область

САПР – система автоматизированного проектирования

СУБД – система управления базами данных

СФПК – система формирования профиля компетенций проектировщика

УМ – учебный материал

ФГОС ВПО – Федеральный государственный образовательный
стандарт высшего профессионального образования

ЭВМ – электронная вычислительная машина

Введение

Актуальность темы. Системы автоматизированного проектирования (САПР) занимают важное место на современном производстве, и требуют наличия у проектировщиков развитых компетенций автоматизированного проектирования [120].

С увеличением сложности современных САПР объектов машиностроения, необходимостью повышения качества проектных решений, ростом сложности технологических решений все более важным становится создание эффективных средств и методов постоянного повышения квалификации проектировщиков.

Однако современные обучающие системы не учитывают специфику обучения проектной деятельности, отсутствует интеграция с профилем компетенций проектировщика и оценка его проектной деятельности. Формирование необходимых компетенций и рекомендаций для проектировщика позволит повысить эффективность проектной деятельности, что обуславливает актуальность задачи исследования и разработки.

Целью исследований является повышение эффективности деятельности проектировщика в системах автоматизированного машиностроительного проектирования за счет формирования необходимых компетенций и рекомендаций для проектировщика.

Задачи исследования

1. Провести анализ современных методов, моделей и средств автоматизированного обучения, механизмов персонификации процесса обучения с целью определения подходов для минимизации суммарного времени обучения с учётом практической подготовки обучаемого, лёгкости наполнения предметной области, наличия материалов для дополнительного изучения.

2. Провести анализ подходов построения систем рекомендаций.

3. Разработать модели автоматизированной обучающей системы: предметной области, профиля проектировщика, сценария обучения.

4. Разработать метод формирования персонифицированного сценария обучения на основе профиля проектировщика, включающий как теоретическую,

так и практическую подготовку, и учитывающий его навыки и умения работы в САПР КОМПАС.

5. Разработать метод формирования рекомендаций для проектировщика на основе протокола проектных операций трехмерного моделирования деталей, выполняемых в САПР КОМПАС.

6. Реализовать предложенные модели, методы и алгоритмы в виде программного комплекса.

Объектом исследования является организация компьютерного обучения автоматизированному проектированию объектов машиностроения на основе пакета САПР КОМПАС.

Предметом исследования являются модели, методы и средства персонифицированного обучения проектной деятельности в САПР КОМПАС.

Методы исследования основаны на использовании положений и методов теории алгоритмов, теории множеств, теории графов, теории автоматов, теории автоматизированных обучающих систем, теории классификации, теории автоматизированного проектирования, основ системотехники.

Научная новизна.

1. Онтологическая модель предметной области, отличающаяся введением классов, атомов и функций, ориентированных на предметную область автоматизированного машиностроительного проектирования, и позволяющая персонифицировать процесс обучения.

2. Метод формирования персонифицированного сценария обучения, отличающийся использованием динамических механизмов взаимодействия моделей пространства обучения (профиля проектировщика, теста, проектно-практического здания) с онтологической моделью предметной области автоматизированного машиностроительного проектирования и обеспечивающий формирование персонифицированных компетенций проектировщика. Предложена теоретическая оценка сокращения времени обучения при использовании персонифицированного сценария обучения.

3. Метод формирования рекомендаций и корректировки профиля компетенций проектировщика на основе протокола проектных операций отличающийся анализом операций трехмерного моделирования деталей, выполняемых в САПР КОМПАС, и позволяющий сформировать рекомендации и скорректировать профиль компетенций проектировщика.

Личный вклад. Научные результаты, приведённые в диссертационной работе и сформулированные в положениях, выносимых на защиту, получены автором лично. Научному руководителю принадлежит формулировка концепции решаемой проблемы и постановка цели исследования. Подготовка к публикации некоторых результатов проводилась совместно с соавторами, причем вклад диссертанта был определяющим.

Практическая ценность полученных результатов состоит в разработке наукоёмкого программного обеспечения, включающего следующие компоненты.

1. Архитектуру интеллектуальной автоматизированной обучающей системы.
2. Графический редактор модели предметной области на языке программирования Java, обеспечивающего хранение данных в системе управления базами данных MySQL и наполнение онтологической модели предметной области.
3. Конструктор тестов на основе предложенной модели тестового задания.
4. Алгоритм формирования персонифицированного сценария обучения на основе предложенной онтологической модели предметной области на языке программирования Java, и позволяющий сократить время обучения на 14%.
5. Базу данных для хранения и обработки компонентов предметной области на базе MySQL-сервера и на основе предложенной онтологической модели.
6. Web-ориентированную обучающую систему на основе предложенных онтологической модели предметной области, профиля проектировщика, модели тестового задания и метода формирования персонифицированного сценария обучения.
7. Мобильную систему обучения для платформы Android на основе предложенных онтологической модели предметной области, профиля проектировщика, модели тестового задания.

8. Алгоритм анализа проектных операций и формирования рекомендаций на основе метода формирования рекомендаций и корректировки профиля компетенций проектировщика, и позволяющий сократить количество выполняемых действий в среднем на 4,4%.

9. Виртуальную компоненту автоматизированной обучающей системы (АОС) на базе автоматного подхода, отличающаяся наличием функциональных блоков и множеством ошибочных состояний, позволяющая формализовать процесс разработки тренажёра, автоматически оценить действия обучаемого на основе шаблонов неэффективных действий.

Апробация работы. Основные положения диссертационной работы докладывались и обсуждались на следующих международных и всероссийских конференциях: Всероссийской конференции с элементами научной школы для молодежи "Проведение научных исследований в области обработки, хранения, передачи и защиты информации", г. Ульяновск, 2009; Российской конференции "Информатика и вычислительная техника" (ИВТ-2010), г. Ульяновск, 2010; Российской школе-семинаре «Информатика, моделирование, автоматизация проектирования» (ИМАП-2010), г. Ульяновск, 2010; Российской конференции "Информатика и вычислительная техника" (ИВТ-2011), г. Ульяновск, 2011; Российской школе-семинаре «Информатика, моделирование, автоматизация проектирования» (ИМАП-2010), г. Ульяновск, 2011; Первой всероссийской научной конференции с международным участием «Системный анализ и семиотическое моделирование (SASM' 2011)», г. Казань, 2011; Российской школе-семинаре «Информатика, моделирование, автоматизация проектирования» (ИМАП-2013), г. Ульяновск, 2013; Научно-технической конференции профессорско-преподавательского состава (ППС-2013), г. Ульяновск, 2013; Всероссийском конкурсе научно-технического творчества молодежи (НТТМ-2014), г. Москва; Молодежном инновационном форуме ПФО – 2015, г. Ульяновск; Шестой международной конференции «Системный анализ и информационные технологии» (САИТ-2015), г. Светлогорск.

Публикации. По теме диссертации опубликовано 29 печатных работ, в том числе 3 статьи в российских рецензируемых научных журналах и 1 статья в издании, индексируемом в WEB OF SCIENCE. Получено 7 свидетельств об официальной регистрации программы для ЭВМ.

Глава 1. Компьютерные системы формирования профиля проектировщика в процессах автоматизированного проектирования машиностроительных объектов

Целью данной главы является исследование методологий, методов и средств формирования профиля проектировщика в процессах автоматизированного проектирования машиностроительных объектов. Определены компетенции проектировщика в соответствии с ФГОС ВПО. Проведён анализ и классификация моделей, методов и средств адаптивного управления процессом обучения в автоматизированных обучающих системах (АОС), рекомендательных систем, подходов к проектированию виртуальных компонентов в АОС. Представлена обобщенная схема формирования профиля компетенций проектировщика машиностроительных объектов.

1.1. Профиль компетенций проектировщика машиностроительных объектов в условиях применения САПР

В списке ключевых терминов образовательных стандартов (ФГОС), разработанных Российской академией образования по поручению Министерства образования и науки Российской Федерации, компетенция определяется как совокупность определенных знаний, умений и навыков, в которых человек должен быть осведомлен и иметь практический опыт работы [107].

Определим понятия знания, умения и навык проектировщика [80].

Знания - совокупность формализованных сведений определенного объема, образующих целостное представление объекта проектирования на определенном уровне или описании определенного этапа автоматизированной проектной деятельности, предъявляемых субъекту в том или ином виде в ходе процесса обучения.

Умения – способность выполнять проектные действия на основе приобретенных знаний автоматизированного проектирования и проектной практики.

Навыки – владение моторными проектными действиями.

“В ряде научных работ, посвященных компетентностному подходу, умения и навыки объединяются под общим термином «навыковые компоненты компетенций» [142, 118]. Это позволяет учесть неразрывность и общность умений, навыков и необходимых для их формирования и реализации личных качеств” [147].

На рисунке 1.1 показана структура компетенции.



Рисунок 1.1 - Структура компетенции

Профиль проектировщика - это совокупность изменяющегося во времени перечня компетенций и их характеристик, зафиксированная в математической модели компоненты обучаемого АОС. При этом компетенции имеют сложный иерархический порядок (рисунок 1.2).

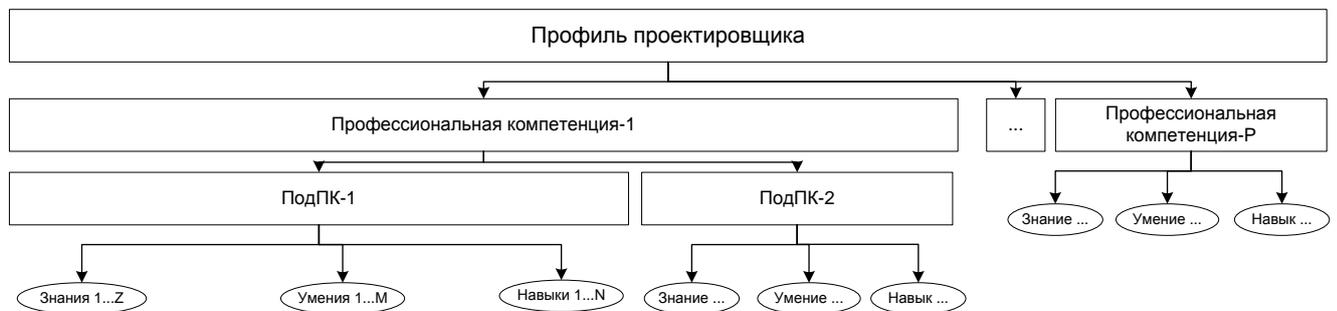


Рисунок 1.2 - Структура профиля компетенций

В рамках работы будем рассматривать компетенции в области конструирования в САПР.

Рассмотрим некоторые существующие стандарты затрагивающие компетенции проектировщиков.

В ФГОС ВПО по направлению подготовки 151900 Конструкторско-технологическое обеспечение машиностроительных производств (квалификация (степень) "БАКАЛАВР") от 2011 года даны следующие подходящие компетенции

[122]: способность использовать современные информационные технологии при проектировании машиностроительных изделий, производств (ПК-11); способность использовать информационные, технические средства при разработке новых технологий и изделий машиностроения (ПК-19). Однако данные компетенции являются общими, в настоящей работе предложены компетенции, носящие детальный и более практический характер.

В проекте профессионального стандарта «СПЕЦИАЛИСТ ПО КОМПЬЮТЕРНОМУ КОНСТРУИРОВАНИЮ» [124] (разработан Министерством труда и социальной защиты Российской Федерации, союзом «Агентство развития профессиональных сообществ и рабочих кадров «Ворлдскиллс Россия») от 2015 года не описаны компетенции, но даны трудовые функции (конструирование, анализ взаимодействия узлов, детальная проработка узлов с учетом внешних факторов), действия, знания и умения инженера-конструктора. Проект стандарта также является достаточно общим для формирования профиля компетенций.

В Российском стандарте профессиональной деятельности инженера-проектировщика [137] разработанным НКО «Ассоциация инженеров «Национальная палата инженеров»» на основе федеральных законов № 169 ФЗ «Об инженерной деятельности в Российской Федерации» от 2015 года, список компетенций находится в разработке.

В квалификационном справочнике должностей руководителей, специалистов и других служащих [123] представлены должностные обязанности и знания инженера-конструктора, в том числе он должен знать системы и методы проектирования, средства автоматизации проектирования.

В перечне компетенций «Перечень профессий (компетенций) WorldSkills Russia» [138], разработанным союзом «Агентство развития профессиональных сообществ и рабочих кадров «Ворлдскиллс Россия» от 2015 года, описана компетенция «Инженерная графика САД» в том числе со следующими навыками: знание и понимание трехмерного моделирования деталей, создание расчётов, обратное конструирование физической модели, технические чертежи и задание размеров.

В сертификационных тестах компании «АСКОН» для уровня М2 «Трёхмерное моделирование деталей и сборочных единиц в системе КОМПАС-3D» определены следующие области знаний [139]: интерфейс и сервисные функции, эскиз, твердотельное моделирование, листовые детали, переменные и параметризация, массивы, пространственные кривые и точки, вспомогательная геометрия; сборка, чертёж, спецификация.

Рассмотрев представленные стандарты, были выработаны следующие компетенции первого уровня:

- 1) умеет работать с интерфейсом;
- 2) владеет навыками построения эскизов;
- 3) владеет навыками твердотельного моделирования;
- 4) способен работать с листовыми деталями;
- 5) умеет работать с переменными и параметризацией;
- 6) способен создавать массивы объектов;
- 7) способен строить пространственные кривые и точки;
- 8) способен применять вспомогательную геометрию;
- 9) владеет навыками проектирования сборки;
- 10) владеет навыками создания чертежей;
- 11) способен работать со спецификацией.

1.2. Методы и средства адаптивного управления процессом обучения в автоматизированных обучающих системах для формирования профиля проектировщика

Основной целью адаптивных обучающих систем является реализация управления процессом обучения с учётом индивидуальных особенностей пользователей. Адаптивные методы позволяют сократить время и повысить эффективность процесса обучения за счёт удержания пользователей в оптимальной зоне обучения [151].

Рассмотрим некоторые адаптивные обучающие системы.

В работе [148] Шабалиной О. А. предметная область представляет собой множество элементов, связанных отношением порядка. В качестве элементов курса могут выступать блоки теоретического материала (лекции, и/или их фрагменты), практические задания, тесты различного вида, упражнения и т.д. Модель обучаемого представляет собой оверлейную модель, состояние обучаемого является отображением множества элементов на множество оценок освоенности этих элементов. В процессе изучения происходит разбиение пространства обучения на три области: область освоенности, неосвоенная область и область, доступная для освоения. Управление процессом обучения заключается в организации такой деятельности, при которой достигается состояние освоенности пространства знаний, удовлетворяющее цели обучения [149]. Фрагмент процесса освоения знаний показан на рисунке 1.3.

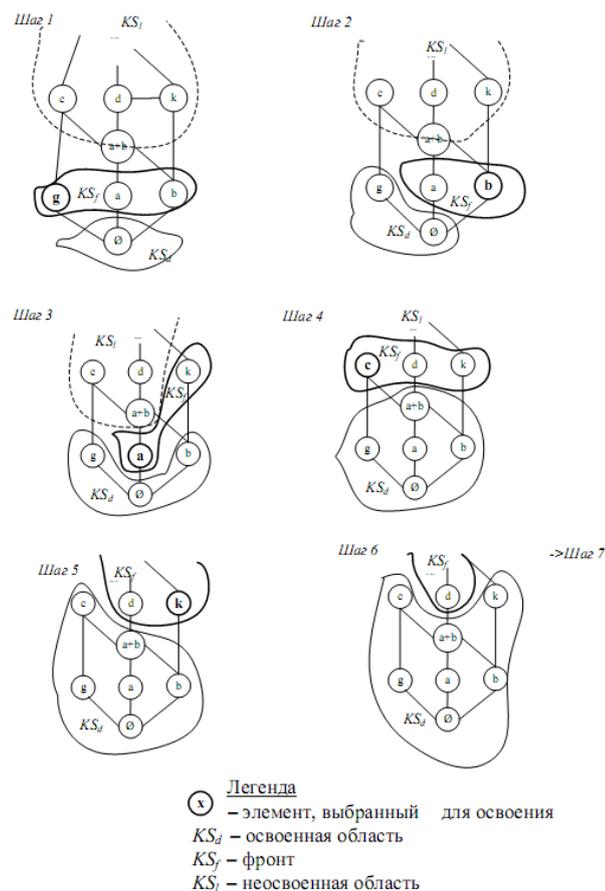


Рисунок 1.3 - Фрагмент процесса освоения пространства знаний [148]

В системе IDEA [113, 95, 58, 32] модель предметной области представлена в виде деревьев, которые содержат учебный материал, тренировочные упражнения,

справочную информацию. Модель обучаемого представлена как база данных истории обучения, в которой содержится информация обо всех событиях, произошедших в процессе обучения. Модель процесса обучения построена на основе экспертной системы, которая корректирует поведение ученика в процессе обучения. Основой для принятия решения служат продукции, оперирующие следующей информацией: текущая стратегия (задача обучения); база данных истории обучения; тип ученика; локальные события; дерево учебных целей. В задачи экспертной системы по управлению обучением входит: «подбор подходящего учебного плана; предъявление ученику учебного материала, накопление результатов урока; составление протокола обучения, моделей ученика и учебной ситуации; анализ ситуации; подбор подходящей учебной стратегии (рисунок 1.4); составление плана следующего урока — наполнение выбранной стратегии учебным материалом» [113].

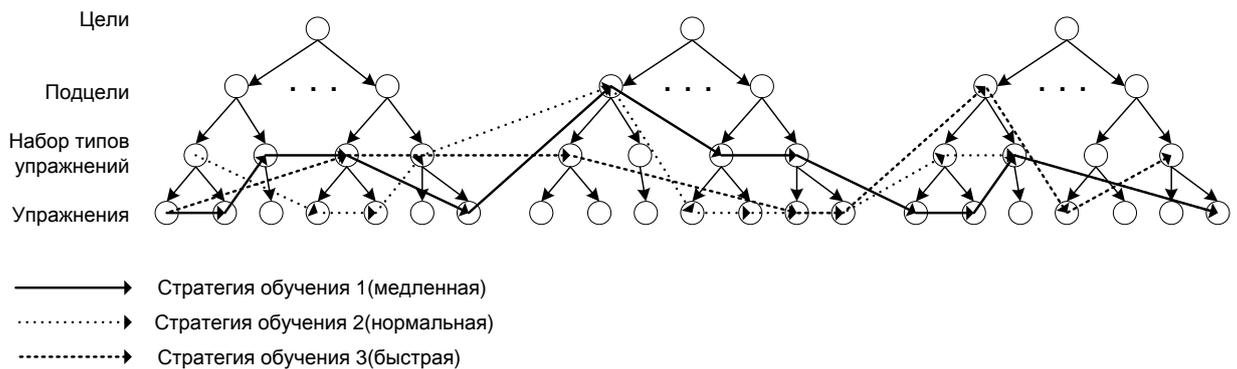


Рисунок 1.4 - Обучающие стратегии в экспертной системе программы «IDEA» [113]

В системе WARE модель предметной области представлена множеством элементов знаний, связанных иерархическими и порядковыми отношениями [104, 103]. Элементы знаний имеют ссылки на ресурсы курса (элементарные информационные ресурсы учебников и задачников, а также примеры, тесты и проекты), содержание которых относится к данной единице знаний. Модель обучаемого представляет собой оверлейную модель, содержащую степени владения знаниями и навыками. Для прохождения определённой темы система позволяет генерировать сценарий обучения на основе сравнения текущего уровня

знаний обучаемого с необходимым для понимания рассматриваемой темы. Алгоритм проверяет, все ли предварительные знания достаточно усвоены студентом. В противном случае алгоритм находит единицы знаний, нуждающиеся в изучении. После этого генерируется последовательность подцелей и последовательность информационных элементов для изучения.

В работе Доррер А.Г. модель предметной области представляется как последовательность модулей с различной сложностью. Каждый модуль содержит учебные и контрольные материалы. Модель обучаемого представляет стереотипную модель со следующими категориями обучаемых: «студенты очной пятигодичной формы обучения; студенты очной ускоренной формы обучения; студенты заочной формы обучения» [90]. Модель процесса обучения построена на основе цветных сетей Петри (рисунок 1.5). В зависимости от уровня ответов обучаемого возможны несколько вариантов обучения: переход к следующему модулю с повышением уровня сложности; изучение дополнительного материала; изучение материала с пониженным уровнем сложности.

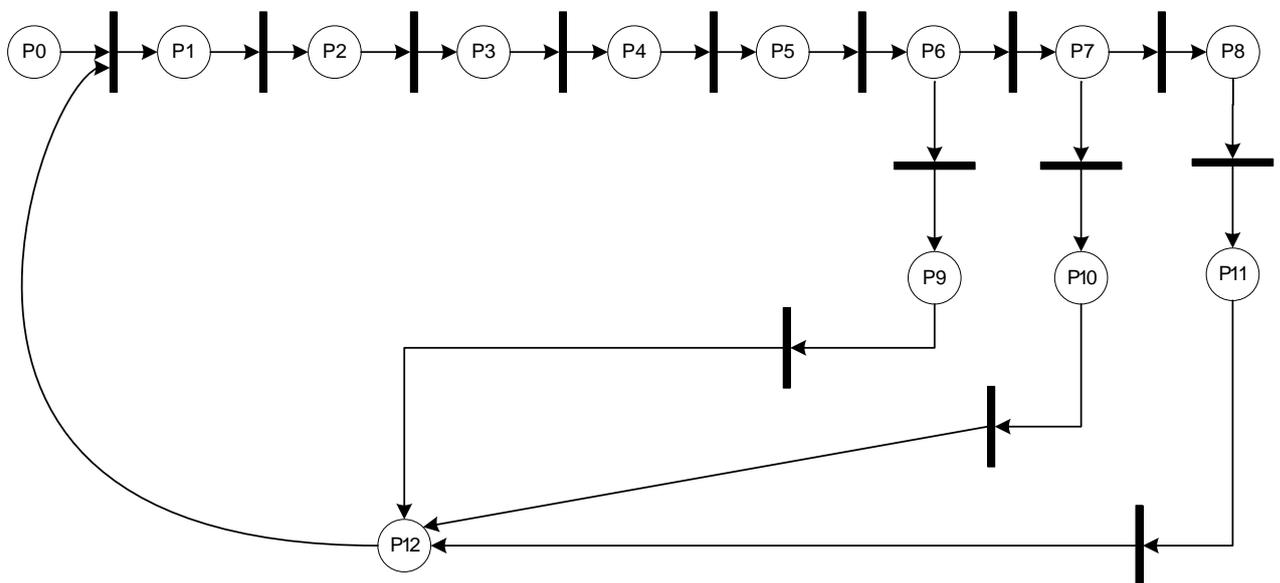


Рисунок 1.5 - Сеть Петри, моделирующая прохождение курса [90]

В работе [31] модель предметной области рассматривается как множество учебных курсов. Для построения оптимального сценария обучения используется генетический алгоритм, где генотипом является последовательность отобранных учебных курсов. В качестве оптимального размера начальной популяции выбрано

50, большая численность увеличит вероятность найти решение высокого качества, но приведет к уменьшению скорости поиска. Функция приспособленности учитывает результаты предварительного тестирования, степень связанности учебных материалов, сложность и изученность материалов. В качестве метода отбора используется метод рулетки, операция мутации выполняется через изменение последовательности учебных планов.

В работе [35] описывается система автоматических обучающих персональных рекомендаций. Модель предметной области представляет собой множество неструктурированных учебных материалов, которая автоматически формируется из сети Интернет с помощью технологий индексации и интеллектуального анализа текста на основе программного обеспечения Nutch. Модель обучаемого содержит взвешенную последовательность обучающих элементов, которые интересны студенту. Обучаемые группируются на основе сходства и различий своих предпочтений, для этого могут использоваться различные методы кластеризации. Рекомендации строятся на основе проявленных интересов через анализ посещённых веб-страниц, используя фильтрации на основе содержимого и коллаборативную. Результаты ранжируются в соответствии со сходством содержимого (мера TF-IDF).

В работе [21] модель предметной области делится на три части: структура, организованная в виде дерева (курс, главы, разделы и подразделы); учебные материалы и тестовые задания. Модель обучаемого представляет собой оверлейную модель, которая инициализируется после предтеста. Для анализа результатов применяется теория Демпстера — Шафера. Уровень знания описывается тремя вариантами: низкий, средний и мастер. Адаптация происходит после теста, система вычисляет уровень знаний и обновляет модель обучаемого. Если студент проходит по оценке текущую главу, ему предлагаются ссылки на следующую, которая должна иметь уровень знания больше или равное общему уровню знаний студента. Если студент не набрал достаточной оценки, система предлагает ссылки на разделы текущей главы, которые необходимо изучить. Система разработана с использованием всех методов адаптивной навигации

(полное руководство, адаптивная сортировка, сокрытие ссылок, аннотирование ссылок).

В работе [42] модель предметной области представлена в виде графа, где узлы являются понятиями предметной области. Предметная область разделена на две подмодели знаний и ресурсов. Подмодель знаний содержит несколько уровней понятий в виде дерева: составное понятие, узел, атомарное понятие, а подмодель ресурсов - несколько версий материала (текстовая версия, аудио версия, визуальная версия, кинетическая версия) (рисунок 1.6). Несколько версий ресурса позволяют подстраиваться под наиболее предпочитаемый стиль обучения студента. Модель обучаемого состоит из трёх компонентов: профиль обучаемого - содержит персональные характеристики обучаемого; модель знаний - оверлейная модель над понятиями предметной области; предпочитаемый стиль обучения. Модель адаптации строится на основе продукции и разделена на два слоя: выбор понятий из пространства знаний предметной области для изучения и выбор способа представления материала. Система использует сочетание техник адаптивной навигации и адаптивного представления.

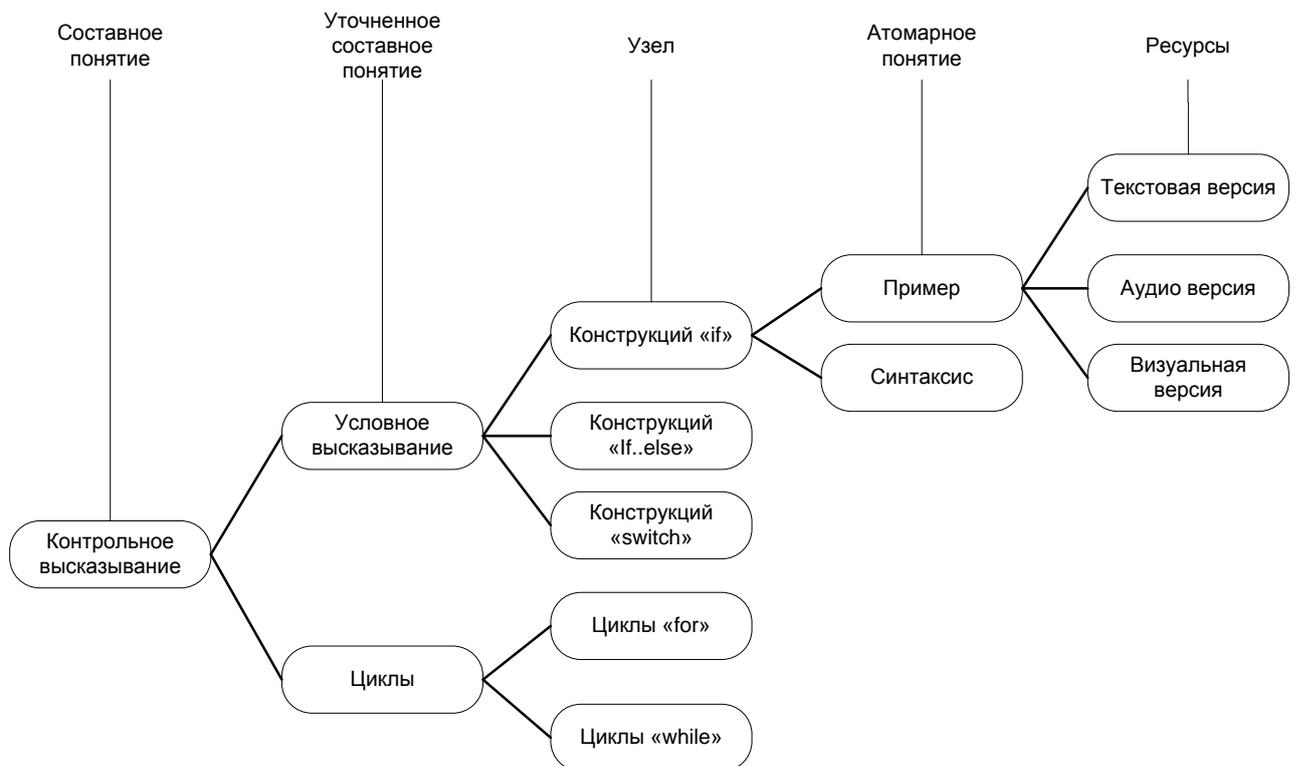


Рисунок 1.6 - Пример предметной области [42]

В системе AEHS-TLS [39] учитывается стиль мышления обучаемого. Модель предметной области содержит 12 типов материалов: 8 теоретических и 4 практических. Модель обучаемого описывается 3 субмоделями: цели и предпочтения (какие курсы хочет посетить, каковы предпочтения, такие как шрифт, тип, размер, цвет и другие параметры, связанные с интерфейсом); мышление и стиль обучения (содержит информацию о конкретном пути обучении и подходе к обучению); знание и исполнение (результаты тестов, проектов, задач). Система поддерживает следующие стили мышления: теоретик, организатор, новатор, гуманный. Стили обучения: активный, рефлексивный, дедуктивный, индуктивный, визуальный, словесный, последовательный, глобальный. Модель процесса обучения строит наиболее подходящий сценарий обучения с использованием адаптивной навигации, выбором содержания и аннотированных ссылок, которые согласуются с профилем ученика. Сценарий обучения выбирается на основе формулы, которая учитывает стиль мышления, стиль обучения и уровень знаний.

В системе QuizGuide [51] модель предметной области структурирована в упорядоченное множество тем и целей (рисунок 1.7). Однако, в случае необходимости, сложная тема может быть разделена на несколько целей. С темами связано множество викторин. Модель обучаемого - оверлейная, знания студента представлены в виде наложения на предметную область (для каждой темы система поддерживает оценку знаний студентов в этой теме). Подсистема управления процессом обучения направляет студентов в наиболее подходящие темы, представляя адаптивную навигацию. Каждая тема помечается значком в соответствии отношения темы к текущей цели обучения и уровню знаний, продемонстрированные в процессе обучения. Используя достаточно простую модель предметной области, авторы пытались достичь компромисса между сложностью организации моделей предметной области и обучаемого, с одной стороны, и способностью учителей структурировать содержание и использовать адаптационные технологий, с другой.

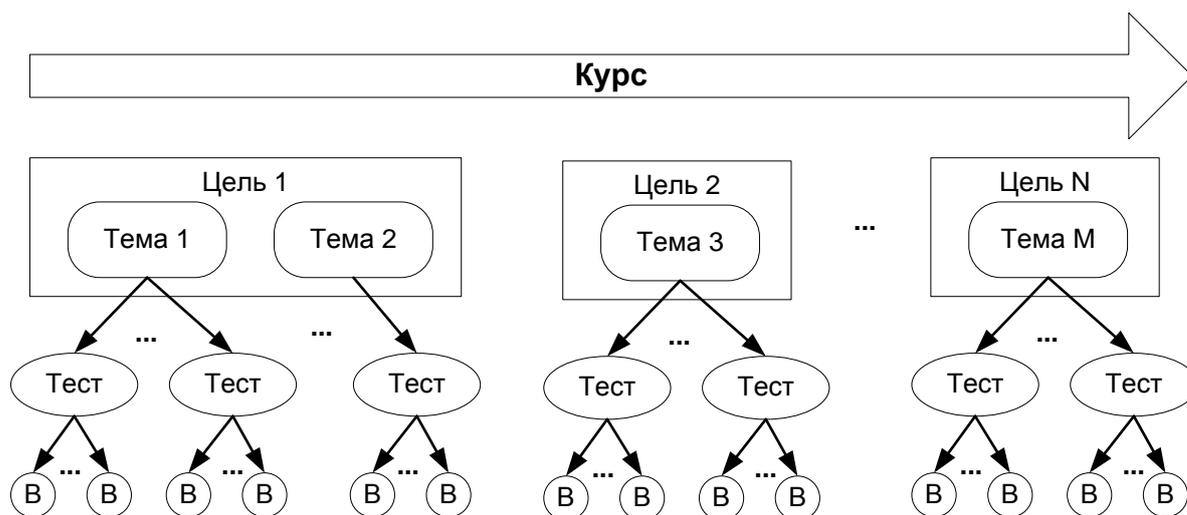


Рисунок 1.7 - Модель Предметной области [51]

В системе GRAPPLE [49, 29] модель предметной области описывается через понятия и отношения, с понятиями могут быть связаны информационные ресурсы (которые описывают концепцию: изображение, текст и другое) и факты (свойства концепции). Модель обучаемого - оверлейная. Модель адаптации построена на основе множества правил “если-то”, использует модели предметной области и обучаемого и принимает решение, какую информацию и как отобразить (адаптивное представление, адаптация на уровне содержания, адаптация на уровне ссылок). В рамках проекта разработан язык GAL (Generic Adaptation Language).

Компания DreamBox Learning, основанная в 2006 году, предоставляет программное обеспечение, которое фокусируется на математическом образовании начального и среднего школьного уровня по принципу "обучение на практике"[7]. Система, имея данные о миллионах учениках, использует алгоритмы интеллектуального анализа данных для сегментации и прогнозирования, определяет какая последовательность уроков, какие подходы являются наиболее эффективными для данного студента (рисунок 1.8). Система анализирует все движения мыши и тайминги (собирает около 60 поведенческих значений). Подстраивает под обучаемого следующие параметры: последовательность уроков, навигацию, содержание, темп, нагрузку [19, 12]. Основная задача – держать обучаемого в оптимальной зоне обучения. Также поддерживается система

подсказок, которые помогают пересмотреть выбранные стратегии и решения. Модель предметной области представлена множеством уроков.

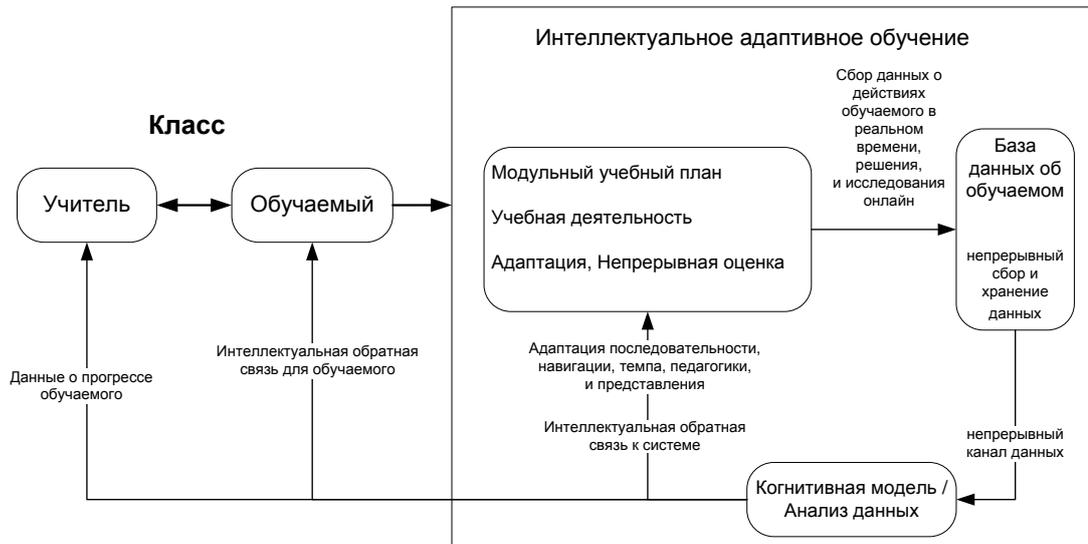


Рисунок 1.8 - Система адаптации обучения

В системе РСМАТ [9, 47, 48] модель предметной области представлена в виде графа понятий. С понятиями связаны учебные объекты - любые информационные ресурсы. С каждым учебным объектом связан файл метаданных, содержащий такие данные, как название, краткое описание и ключевые слова, дата создания, тип файла и другую техническую информацию; информацию об авторских правах; образовательные метаданные - возрастная группа обучаемых, сложность. Модель обучаемого состоит из предметно-зависимых (рисунок 1.9) и предметно-независимых данных. Предметно-зависимая часть является оверлейной моделью (предполагается, что знания студента представляет собой подмножество системных знаний), и включает в себя учебные цели, план обучения, полное описание последовательности обучения, уровень знаний. Предметно-независимые данные могут быть двух типов: общие - описание физического лица, образование, навыки, базовые знания; психологические - стиль обучения, познавательные способности, описание личности и т.д. Для представления модели также используется стереотипная древовидная модель. Процесс обучения начинается с идентификации подгруппы, к которой принадлежит обучаемый с помощью анкеты. Стереотипы используются для адаптации информации, интерфейса, сценария, целей и планов обучения. Модель управления процессом обучения построена на

правилах адаптации. Правила основаны на поведении студента, на знаниях, и на стиле обучения. Учебный план устанавливается учителем, но настраивается с помощью платформы. Правила адаптации отрабатывают следующую функциональность: проверку доступа студента к контенту; обновление стиля обучения и уровня знаний; отображение информации в соответствии со знаниями и стилем обучения; регулирование сценария обучения; контролирование структуры гиперссылок.

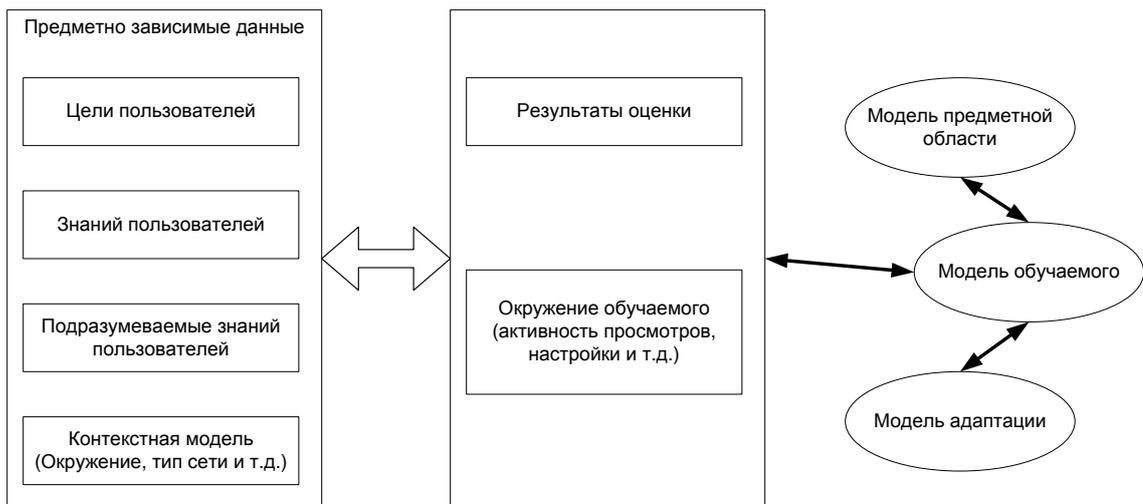


Рисунок 1.9 - Архитектура предметно зависимых данных [29]

Таким образом в АОС выделяют три основных модели: модель предметной области [94], модель обучаемого и модель процесса обучения. Наиболее распространёнными типами моделей предметной области являются [102, 72, 81]:

- продукционные,
- фреймовые,
- формальные логические,
- на основе онтологий,
- семантические сети,
- древовидные,
- множество.

Модели обучаемого:

- оверлейная,
- векторная,

- скалярная,
- стереотипная.

Методы адаптации:

- авторский алгоритм,
- экспертная система [87],
- на основе правил,
- генетический алгоритм [83].

Важнейшей составляющей компетентности специалиста, кроме теоретических знаний, является умение применять эти знания в работе. Современные производства накопили большой опыт решений определённых задач, к сожалению, большинство обучающих систем не используют реальные задачи при подготовке специалистов, а также не интегрирует базу проектных решений предприятия. Из рассмотренных решений только система Khribi, M. K., Jemni, M., & Nasraoui позволяет наполнять предметную область из любых источников Интернет, в том числе из информационной среды предприятия.

В таблице 1.1 показаны сравнительные характеристики представленных систем. Объясним некоторые параметры таблицы. Параметр «Альтернативный сценарий обучения» - определяет возможность предоставлять различный материал для объяснения одной и той же темы, что делает процесс обучения более разнообразным и интересным. Параметр «Вспомогательный материал» определяет возможность использования различных вспомогательных средств при возникновении проблем в процессе обучения: системы подсказок, ссылок на дополнительные обучающие материалы, справочники и т. д. «Лёгкость наполнения» характеризует трудоёмкость работы при составлении курсов. Среди рассмотренных систем можно выделить 3 уровня сложности: простая – полностью автоматическое построение курса (рекомендательная система Khribi, M. K., Jemni, M., & Nasraoui), средняя – наполнение с помощью авторских редакторов, сложная – системы на основе правил, в которых необходимо разрабатывать продукции для обеспечения адаптивной составляющей.

Таблица 1.1 - Сравнительные характеристики автоматизированных обучающих систем

Система	Год	Модель предметной области	Модель обучаемого	Метод адаптации	Интеграция с базой проектных решений	Практическая подготовка	Формирование рекомендаций при практической подготовке	Альтернативный сценарий обучения	Вспомогательный материал	Лёгкость наполнения
Авторская система Шабалиной	2013	Ориентированный граф	Оверлейная	Авторский алгоритм	-	-	-	-	-	+
IDEA	2009	Древовидная	Векторная	Экспертная система	-	-	-	+	+	-
WARE	2006	Семантическая сеть	Оверлейная	Проверка уровня знаний обучаемого для доступа к обучающим материалам	-	+/-	-	-	+/-	+
Авторская система Доррер	2007	Древовидная	Стереотипная	Проверка уровня знаний обучаемого для доступа к	-	-	-	-	+/-	+

Система	Год	Модель предметной области	Модель обучаемого	Метод адаптации	Интеграция с базой проектных решений	Практическая подготовка	Формирование рекомендаций при практической подготовке	Альтернативный сценарий обучения	Вспомогательный материал	Лёгкость наполнения
				обучающим материалам, формирование множества плохо освоенного материала						
Авторская система Huang, M.J., Huang, H.S., Chen, M.Y	2007	Множество	Векторная	Генетический алгоритм	-	-	-	-	-	+
Авторская система Khribi, M. K., Jemni, M., & Nasraoui	2009	Множество	Векторная и стереотипная	Коллаборативная фильтрация, фильтрации содержимого	+/-	-	-	-	+	+

Система	Год	Модель предметной области	Модель обучаемого	Метод адаптации	Интеграция с базой проектных решений	Практическая подготовка	Формирование рекомендаций при практической подготовке	Альтернативный сценарий обучения	Вспомогательный материал	Лёгкость наполнения
Авторская система V. Esichaikul, S. Lamnoi, and C. Bechter	2011	Древовидная	Оверлейная	Проверка уровня знаний обучаемого для доступа к обучающим материалам, формирование множества плохо освоенного материала	-	-	-	-	-	+
Авторская система Yasir Eltigani Ali Mustafa and Sami Mohamed Sharif	2011	Древовидная	Оверлейная	На основе правил	-	-	-	+	-	-
AEHS-TLS	2013	Множество	Векторная	Авторский алгоритм	-	-	-	-	+	-

Система	Год	Модель предметной области	Модель обучаемого	Метод адаптации	Интеграция с базой проектных решений	Практическая подготовка	Формирование рекомендаций при практической подготовке	Альтернативный сценарий обучения	Вспомогательный материал	Лёгкость наполнения
QuizGuide	2015	Древовидная	Оверлейная	Проверка уровня знаний обучаемого для доступа к обучающим материалам, формирование множества плохо освоенного материала	-	-	-	-	+/-	+
GRAPPLE	2009	Семантическая сеть	Оверлейная	На основе правил	-	-	-	-	+	-
PCMAT	2013	Семантическая сеть	Оверлейная, стереотипная	На основе правил	-	-	-	+	+	-

1.3. Анализ систем рекомендаций для проектировщиков

Основная задача системы рекомендации – сформировать список объектов наиболее интересных для пользователя на основе информации о его профиле; текст, помогающий решить задачу более оптимальным методом, выбрать подходящий вариант решения задачи, улучшить свои компетенции и т.д.; экземпляр проектного решения и др. Системы рекомендации преимущественно построены на основе следующих методов [85]:

- 1) методы коллаборативной фильтрации [41, 73],
- 2) методы фильтрации содержимого [46, 37],
- 3) методы, основанные на знаниях [24, 11, 4].

Методы коллаборативной фильтрации вырабатывают рекомендации, основанные на модели предшествующего поведения пользователя. Эта модель может быть построена исключительно на основе поведения данного пользователя или — что более эффективно — с учетом поведения других пользователей со сходными характеристиками.

Основная идея - использовать информацию о прошлом поведении сообщества пользователей для предсказания, какие объекты могут заинтересовать текущего пользователя. Эти типы систем широко распространены, в частности, в качестве инструмента для интернет-сайтов розничной торговли, чтобы определить потребности конкретного клиента и, тем самым, стимулировать дополнительные продажи.

Данный метод принимает матрицу данных оценок пользовательских объектов на вход и, как правило, формирует следующие результаты:

- прогноз, указывающий, в какой степени текущему пользователю подходит определенный объект;
- перечень рекомендуемых объектов.

Методы контентной фильтрации используют содержимое объектов для получения рекомендаций. Данные методы основаны на принципе поиска объектов, схожих с теми, которые пользователь предпочитал в прошлом. Несмотря на то, что

такой подход должен опираться на дополнительные сведения об объектах и пользовательских предпочтениях, он не требует наличия большого пользовательского сообщества или истории рейтинга - списки рекомендаций могут быть сгенерированы, даже если есть только один единственный пользователь.

Методы, основанные на знаниях, ищут подходящие объекты на основе пользовательских требований и своей базы знаний.

Большой вклад в область разработки методов построения рекомендательных систем внесли Ken Goldberg, Upenra Shardanand, Pattie Maes, John S. Breese, Peter W. Foltz, Susan T. Dumais, Jonathan L. Herlocker, Robin Burke, Marko Balabanović, Yoav Shoham.

Рассмотрим некоторые рекомендательные системы.

SUGGEST

SUGGEST — это рекомендательный механизм типа Top-N [18, 52] реализованный в виде библиотеки. Механизм SUGGEST, созданный Джорджем Кариписом в Университете Миннесоты, использует несколько алгоритмов коллаборативной фильтрации и реализует коллаборативную фильтрацию на основе пользователей и на основе элементов. Используется для рекомендации продуктов клиенту, которые, скорее всего он купит; подходящие фильмы, телевизионные программы или музыку; интересные веб-страницы; или альтернативные способы поиска информации.

Преимуществами SUGGEST являются [52].

1. Высокое качество рекомендаций. На широком диапазоне данных алгоритм на основе элементов даёт результаты до 30% лучше по сравнению с традиционными алгоритмами коллаборативной фильтрации.

2. Скорость. Алгоритм на основе элементов вычисляет топ-10 менее чем за 5 миллисекунд.

3. Масштабирование. Оба алгоритма на основе пользователей и на основе элементов работают на больших наборах данных без значительного ухудшения эксплуатационных характеристик.

Easyrec

Easyrec [40] является веб-приложением с открытым исходным кодом, обеспечивает персонализированные рекомендации с помощью веб-служб, которые интегрируются в сторонние веб-приложения.

Механизм easyrec предоставляет интерфейс на базе REST (Representational State Transfer), что расширяет возможности по интеграции сервиса со сторонними приложениями и повышает масштабируемость сервиса.

Easyrec предоставляет базовые типы рекомендаций, которые охватывают наиболее распространенные случаи, встречающиеся во многих сценариях прогнозирования:

- «элементы, связанные с данным элементом»,
- «другие пользователи также просмотрели эти элементы»,
- «конкретные рекомендации для данного пользователя»,
- «другие пользователи также приобрели данный элемент».

Easyrec имеет несколько преимуществ.

1. Быстрая интеграция с Вашим сайтом с помощью плагинов или технологии REST Application Programming Interface (API).

2. Лёгкое администрирование, сервис берёт на себя обязанности по поддержанию надёжности и скорости работы приложения.

LightFM

LightFM обеспечивает эффективные реализации алгоритма матричной факторизации коллаборативной фильтрации и гибридных моделей с использованием явной и скрытой обратной связи.

Структура модели LightFM [38] отвечает двум требованиям.

1. Модель должна узнавать пользователя и элемент представления из данных взаимодействия: если элементы «бальное платье» и «юбка-карандаш» последовательно понравились пользователям, модель должна сделать вывод, что бальные платья похожи на юбку-карандаш.

2. Модель должна вычислять рекомендации для новых элементов и пользователей.

Первое требование реализовано через скрытое представление. Если бальные платья и юбки-карандаши понравились одним и тем же пользователям, они будут близко расположены друг к другу; если бальные платья и байкерские куртки никогда не нравились одним и тем же пользователям, тогда они будут расположены далеко друг от друга. Таким образом, если представления для бальных платьев и юбок-карандашей похожи, можно с уверенностью рекомендовать бальные платья для нового пользователя, который до сих пор интересовался только юбками-карандашами.

Второе требование реализовано через механизмы, анализирующие содержимое объектов. Элементы и пользователи представлены в виде линейных комбинаций их особенностей контента. Особенности содержания известны в момент добавления пользователей или элементов в систему, что позволяет сразу формировать рекомендации. Например, рекомендации для джинсовой куртки рассматривается как сумма рекомендаций для джинсов и куртки; для пользователей «женщины из США» – сумма пользователей-женщин и пользователей из США.

LightFM имеет несколько преимуществ.

1. Обучение с использованием алгоритмов Bayesian Personalized Ranking, Weighted Approximate-Rank Pairwise, стохастического градиентного спуска.

2. Эффективная реализация параллельной обработки с использованием Cython и OpenMP.

WeeVis

WeeVis [1, 22] разработана исследовательской группой Applied Software Engineering Грацского технического университета, является вики-средой для создания рекомендаций, основанной на знаниях. Рекомендательные системы на основе ограничений помогают пользователям разобраться в сложных продуктах и услугах, таких как цифровые фотоаппараты, компьютеры, финансовые услуги и

муниципальные услуги. Расчет рекомендаций основывается на базе знаний явно определенных правил.

WeeVis находится в свободном доступе в качестве платформы и успешно применяется несколькими австрийскими университетами (в лекциях про рекомендательные системы), в сфере финансовых услуг, а также в области электронного правительства. Система помогает решить следующие задачи.

1. Интернет - консультация для граждан, например, о документах, которые необходимы для частного строительства. Такие рекомендации приведут к сокращению времени для жителей и должностных лиц.

2. Информационная платформа, например, для определения оптимальной стратегии утилизации отходов для домашнего хозяйства. Вместо того, чтобы предоставить информацию как обычную веб-страницу, система подберёт оптимальную стратегию на основе конкретной ситуации.

Система обеспечивает вики-интерфейс пользователя, который позволяет совместно разрабатывать рекомендации. Взамен инкрементального диалога, в котором пользователь отвечает на один вопрос за другим, как CWAdvisor [1], WeeVis предоставляет интегрированный интерфейс, в котором можно свободно отвечать на вопросы в любом порядке. Интерфейс также поддерживает интеллектуальные механизмы для предоставления альтернативных решений в ситуациях, когда не удастся найти решение для данного набора требований, т.е. требований несовместимых с базой знаний. Для каждого решения рассчитывается оценка совпадения. Если решение удовлетворяет всем требованиям, эта оценка составляет 100%, в противном случае она ниже.

База знаний представлена в виде трёх множеств:

$$S = (U, P, C),$$

где U – множество требований, P – множество решений, C – множество ограничений.

VITA

VITA [14, 23] – это рекомендательная система для финансовых услуг, которая была создана кредитной ассоциацией в Венгрии для помощи операторам при общении с клиентами.

Торговые представители в сфере финансовых услуг сталкиваются с проблемой возросшей сложности сервисных решений. Во многих случаях агенты не знают, какие услуги должны предлагаться в каких случаях и как они должны быть объяснены. В связи с этим основной целью является повышение общей производительности торговых представителей (например, увеличение количества контактов или количества продуктов, продаваемых в течение определенного периода времени) и качества самих диалогов при продажах. Достижение этих целей увеличивает заинтересованность клиента в долгосрочных деловых связях с провайдером финансовых услуг.

В соответствии с этим решением определены следующие основные цели: повышение эффективности продаж; эффективное развитие и обслуживание программного обеспечения: новые технологии должны облегчить разработку баз знаний в области продаж.

База знаний состоит из следующих элементов.

1. Свойства клиентов: каждый клиент должен четко сформулировать свои требования, которые являются элементарными условиями для рекомендаций. Примерами в сфере финансовых услуг являются возраст, стаж, существующие кредиты и так далее.

2. Свойства продукта: каждый продукт описывается через набор predetermined свойств, таких как рекомендуемое время работы, предсказанная доходность, ожидаемый риск и так далее.

3. Ограничения, которые определяют, какие продукты должны быть рекомендованы в каком контексте. Простой пример такого ограничения: клиенты, не готовые рисковать должны, получить рекомендации, которые не включают в себя продукты с высокой степенью риска.

4. Сценарий процесса: явное определение последовательности вопросов, которые должны быть заданы пользователю.

Процесс рекомендации кредита структурирован на разных этапах (выявления требований, проверка кредитоспособности, выбор продукта, детальный расчет/представления результатов). На первом этапе запрашивается основная информация о клиенте (персональные данные), определяются основные цели и требования в отношении кредита (сумма кредита, срок выдачи). Следующая задача состоит в том, чтобы проверить кредитоспособность клиента на основе полученной подробной информации о финансовом положении и имеющихся ценных бумагах. В это время рекомендательная система проверяет, может ли найдено быть решение под заданные требования. Если такого решения нет (например, запрошено слишком много денег), приложение попытается определить альтернативы для текущих требований и доступного набора продуктов. После успешного завершения проверки фазы кредитоспособности, предлагаются доступные продукты. После выбора предпочтительного варианта выполняется детальный расчет специфических свойств продукта, таких как размеры ежемесячных погашений.

Система формирования рекомендаций для проектировщика на основе протокола проектных операций должна удовлетворять следующим требованиям:

- расширяемость – система должна позволить эксперту по машиностроительному проектированию добавлять новые рекомендации;
- конкретность решений – рекомендации должны давать конкретные решения для анализируемых ситуаций;
- оценка рекомендаций – оценка результатов, к которым приведёт выполнение полученной рекомендации;
- интеграция с профилем пользователя – корректировка профиля пользователя на основе сформированных рекомендаций, чтобы определить, какая помощь ему потребовалась в процессе работы.

В таблице 1.2 показаны сравнительные характеристики представленных систем.

Таблица 1.2 - Сравнительные характеристики рекомендательных систем

Система	SUGGEST	Easyrec	LightFM	WeeVis	VITA
Используемые методы	Коллаборативной фильтрации	Коллаборативной фильтрации	Коллаборативной фильтрации; анализ содержимого объектов	Основные на знаниях	Основные на знаниях
Формирование рекомендаций	+	+	+	+	+
Использование опыта других участников	+	+	+	-	-
Использование данных об объекте	-	-	+	+	+
Расширение рекомендаций	-	-	-	+	+
Формирование конкретных решений	-	-	-	+	+
Оценка эффективности рекомендаций	-	-	-	-	-
Интеграция с профилем пользователя	-	-	-	-	-

На основе анализа таблицы можно сделать вывод, что не одна из представленных систем не подходит для формирования рекомендаций на основе протокола проектных операций. Также видно, что новый метод формирования рекомендаций должен строиться на основе знаний, только он позволяет расширять рекомендации и формировать конкретные решения.

1.4. Основные подходы к проектированию виртуальных компонентов автоматизированных обучающих систем

Одним из подходов в подготовке инженеров для развития их прикладных компетенций является использование виртуальных тренажёров. Подготовка специалиста на реальном оборудовании часто экономически неэффективна, сопряжена с рисками для здоровья, а иногда и просто не возможна. Успехи компьютерных технологий активизировали бурное развитие виртуальных тренажёров, позволили разрабатывать целые виртуальные лаборатории.

В настоящее время разработка компьютерных тренажеров начало превращаться в самостоятельную отрасль прикладных компьютерных наук, сочетающую в своих изделиях современные компьютерные технологии с научными знаниями из разных дисциплин, которым посвящено множество научно-технических и научно-методических работ [59, 70, 89, 115]. В области виртуальных тренажёров достигнуты значительные успехи [13], связанные с симуляцией реальных объектов и систем, организацией интерфейса с обучаемыми, формированием банка упражнений и заданий, однако задача оценки действий обучаемых в полном объеме не решена.

Вопросами математического обеспечения тренажёрных систем занимаются ученые: Черепашков А. А. [144, 145, 146], Лискин В.А., Красовский А.А. [111], Сергеев С.Ф. [136], Дозорцев В.М. [88]. Основными разработчиками тренажёрных систем в России являются ОАО "Тренажерные системы", ЗАО "Кронштадт", ЗАО "Транзас" и другие.

Выделим ряд основных подходов к проектированию тренажёров [65].

1. Сценарный. В основе работы тренажёра лежит сценарий обучения. В каждой точке сценария предлагаются варианты действий обучаемых. При правильном ответе обучаемый переходит на следующий этап. Пример такого тренажёра рассмотрен в работах [116, 141]. Основным недостатком является трудоёмкость построения сложных тренажёров.

2. На базе автоматного подхода. Модели состояний тренажёра являются дискретными и представляются в виде автоматных моделей:

$$A = (S, D, Fs, Fd),$$

где S - конечное множество параметров, характеризующих состояние объекта;

D - конечное множество управлений;

Fs - функции выходов;

Fd - функции переходов.

Модель в чистом виде практически не используется в силу своей ограниченности, примеры рассмотрены в работе [110].

3. Алгоритмический. Как правило, такие тренажёры строятся для моделирования сложных процессов. В основе лежат математические модели и методы численного анализа [44] для расчёта показаний системы. Для реализации тренажёров используются различные инструментальные среды конечно-элементного анализа для решения задач в области механики, теплообмена, гидродинамики, например, ANSYS [20], ABAQUS [3]. Данное решение особенно популярно при построении компьютерных тренажёров для обучения персонала нефтегазовой отрасли, например, тренажер-симулятор бурения DART, тренажер-имитатор капитального ремонта скважин АМТ-411, тренажер-имитатор эксплуатации и освоения скважин АМТ-601, а также различные авторские тренажёры [117]. Основными недостатками являются трудоёмкость разработки и узкая специализация.

4. Гибридный. Сочетает сильные стороны автоматного и алгоритмического подходов. В основе построения тренажёра лежит автоматный подход с расширением сложной логики за счёт алгоритмического моделирования. Примеры этого класса тренажёров приведены в работе [30].

В таблице 1.3 показаны сравнительные характеристики представленных подходов.

Таблица 1.3 - Сравнительные характеристики подходов к построению виртуальных тренажёров

Характеристики\подход к проектированию тренажёра	Сценарные	На базе автоматного подхода	Алгоритмические	Гибридные
Сложность разработки тренажёра	Очень простая	Простая	Сложная	Средняя
Повторное использование кода программы тренажёра	Есть	Есть	Ограниченное	Есть
Моделирование сложных процессов	Нет	Ограниченное	Есть	Есть
Свободная работа с тренажёром	Нет	Есть	Есть	Есть
Автоматическая диагностика ошибок обучаемого	Ограниченная	Есть	Трудоёмкая и не универсальная	Нет

В настоящее время перспективным является гибридный подход, однако он не решает проблему диагностики ошибок обучаемого.

Рассмотрим следующие методы анализа действий обучаемых.

1. Ручная проверка экспертом успеваемости обучаемых специалистов на основе логов тренажёра. Данный способ наиболее распространён вследствие простоты его реализации, универсальности и высокого уровня анализа успехов обучаемого. Примеры тренажёров данного вида представлены в работе [140].

Основной недостаток - трудоёмкость процесса анализа результатов работы с тренажёром.

2. Автоматическая пошаговая проверка процесса работы с тренажёром. Метод популярен среди систем, построенных на основе сценарных и автоматных подходов. Осуществляется проверка каждого действия обучаемого на соответствие ожидаемому эталонному процессу работы с тренажёром. В случае положительного результата проверки обучаемый переходит на следующий шаг. Такой способ представлен в работе [116]. Преимуществами данного подхода являются простота реализации и универсальность. Недостатки - отсутствие свободы выбора решения у обучаемого, невозможность построения рекомендаций для обучаемого, результат анализа отвечает только на вопрос – правильно / неправильно.

Среди указанных методов наиболее эффективным подходом к диагностике ошибок обучаемого при работе со сложными объектами является экспертная оценка. В основном такая оценка проводится на основе протокола действий или визуального наблюдения. Однако в случае большого количества обучаемых и ограниченного числа экспертов использовать данный метод не представляется возможным.

1.5. Разработка обобщенной схемы формирования профиля компетенций проектировщика в процессах автоматизированного проектирования

Обобщенная схема формирования профиля компетенций проектировщика в процессах автоматизированного проектирования машиностроительных объектов на примере САПР КОМПАС в виде диаграммы IDEF0 [33] приведена на рисунке 1.10.

Механизмами, которые обеспечивают процесс формирования профиля компетенций, являются САПР КОМПАС, система рекомендаций и система обучения.

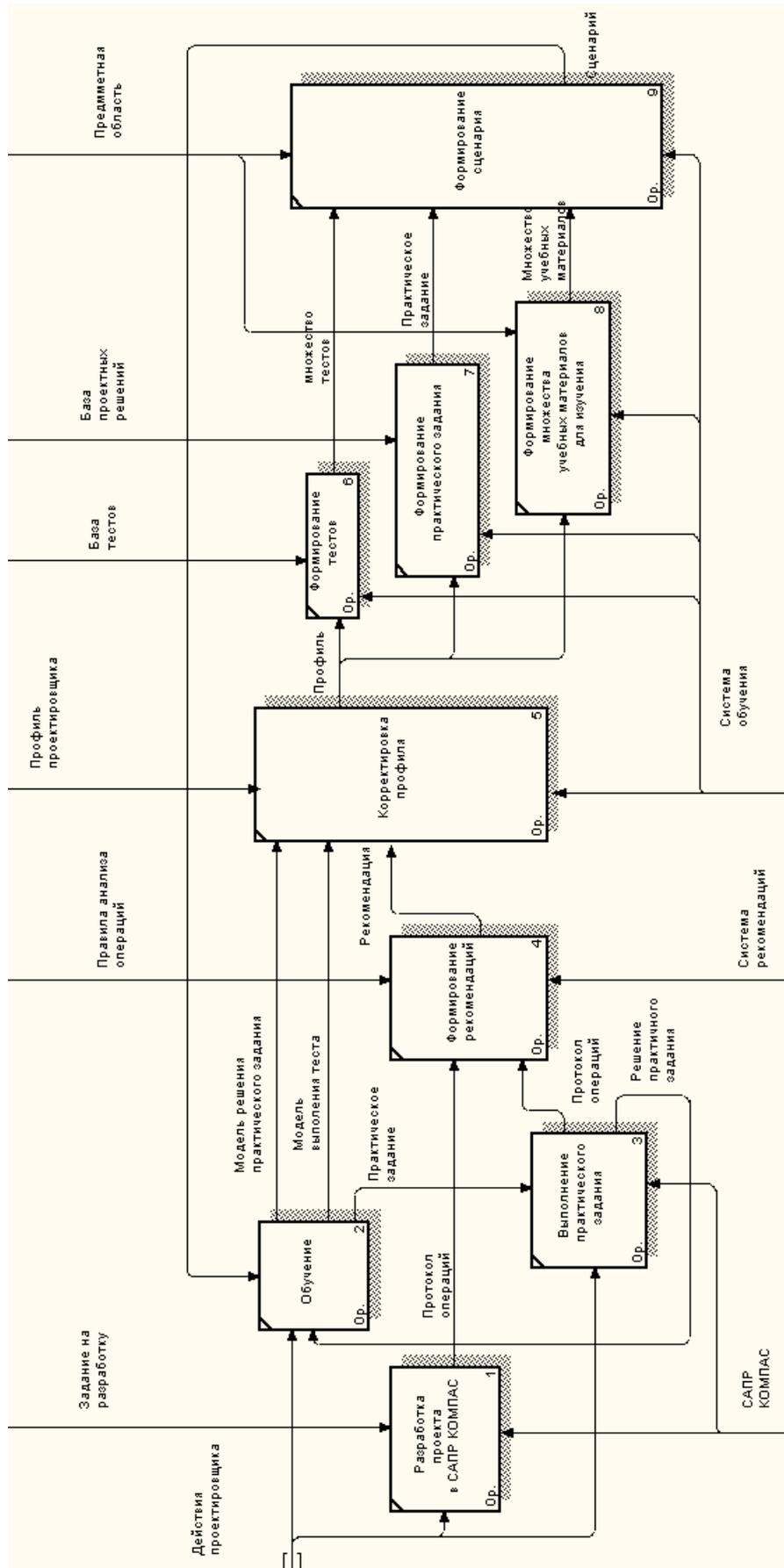


Рисунок 1.10 - Обобщенная схема формирования профиля компетенций проектировщика

В блоке 1 проектировщик на основе задания на разработку выполняет проект в САПР КОМПАС. Протокол выполняемых операций поступает в блок 4 «Формирование рекомендаций», на основе правил анализа операций формируется рекомендация, которая отображается проектировщику (на рисунке не показана в целях увеличения читаемости) и поступает в блок 5. В блоке «Корректировка профиля» изменяется профиль компетенций проектировщика на основе полученных рекомендаций. В случае необходимости запускается процесс формирования сценария обучения с тестами, практическими заданиями и учебными материалами в блоках 6 «Формирование теста», 7 «Формирование практического задания», 8 «Формирования множества учебных материалов», 9 «Формирование сценария».

В блоке 2 происходит обучения проектировщика на основе сформированного сценария, после выполнения теста в блоке 5 корректируется профиль компетенций. Для работы с практическим заданием проектное решение загружается в САПР КОМПАС, в процессе его выполнения также происходит формирование рекомендаций и корректировка профиля компетенций проектировщика.

В соответствии с целью работы были выявлены следующие факторы, влияющие на эффективность деятельности проектировщика:

F_{state} – неоптимальные действия, которые не изменяют состояние проектного решения (отменённые действия),

F_{group} – неоптимальные одиночные действия над групповыми объектами,

$F_{special}$ – неоптимальные действия при построении специальных объектов (для которых существуют специальные операции: фаска, скругление и т.д.),

F_{trans} – неоптимальные действия при переходах между операциями.

1.6. Постановка задачи

Проведенный выше анализ применяемых методологий, методов и средств формирования профиля компетенций проектировщика в процессах автоматизированного проектирования машиностроительных объектов показывает.

1. Обучающие системы не учитывают специфику обучения проектной деятельности, отсутствует интеграция с профилем компетенций проектировщика, с его проектной деятельностью и базой проектных решений предприятия.

2. Эффективные механизмы персонификации процесса обучения требуют трудоёмкой работы по созданию курсов, так как основаны на системах продукций.

3. Отсутствуют механизмы оценки проектировщика в процессе проектной деятельности, формирования рекомендаций и корректировки профиля компетенций.

На основании вышесказанного и проведенного в первой главе данной работы анализа предметной области, сформулируем основные исследовательские задачи.

1. Провести анализ современных методов, моделей и средств автоматизированного обучения, механизмов персонификации процесса обучения с целью определения подходов для минимизации суммарного времени обучения с учётом практической подготовки обучаемого, лёгкости наполнения предметной области, наличия материалов для дополнительного изучения.

2. Провести анализ подходов построения систем рекомендаций.

3. Разработать модели автоматизированной обучающей системы: предметной области машиностроительного проектирования, содержащую теоретические материалы, проектные задания, и позволяющую персонифицировать процесс обучения; профиля проектировщика, содержащую его знания, умения, навыки и компетенции; сценария обучения, состоящую из учебных материалов, тестовых и практических заданий.

4. Разработать метод формирования персонифицированного сценария обучения на основе профиля проектировщика и онтологической модели предметной области, включающий как теоретическую, так и практическую подготовку, и учитывающий его навыки и умения работы в САПР КОМПАС.

5. Разработать метод формирования рекомендаций для проектировщика на основе протокола проектных операций трехмерного моделирования деталей, выполняемых в САПР КОМПАС. Сформированные рекомендации должны скорректировать профиль проектировщика за счёт связи между выполненными

операциями, навыками и умениями проектировщика. Разработанный метод повысит эффективность деятельности проектировщика за счёт поиска не оптимально выполненных проектных операций и рекомендацией замены их на операции с меньшим количеством действий.

6. Реализовать предложенные модели, методы и алгоритмы в виде программного комплекса.

Сформулируем основные гипотезы диссертационного исследования.

1. Если сформировать для обучаемого проектировщика персонализированный сценарий обучения на основе взаимодействия моделей пространства обучения (профиля проектировщика, теста, практического задания) с онтологической моделью предметной области машиностроительного проектирования, а также провести интеграцию с базой проектных решений предприятия, то это повысит скорость и качество обучения проектировщика.

2. Если в процессе работы проектировщика будут сформированы рекомендации на основе анализа выполненных им проектных операций трехмерного моделирования деталей, предлагающие заменить неоптимальные множества операций на множества операций с меньшим количеством действий, то это позволит повысить эффективность работы проектировщика.

1.7. Выводы

1. В рамках диссертации профиль проектировщика включает следующие компетенции: умеет работать с интерфейсом; владеет навыками построения эскизов; владеет навыками твердотельного моделирования; способен работать с листовыми деталями; умеет работать с переменными и параметризацией; способен создавать массивы объектов; способен строить пространственные кривые и точки; способен применять вспомогательную геометрию; владеет навыками проектирования сборки; владеет навыками создания чертежей; способен работать со спецификацией.

2. В автоматизированной обучающей системе для формирования профиля проектировщика выделяют три основные модели: модель предметной области,

модель обучаемого и модель процесса обучения. Наиболее распространёнными типами моделей предметной области являются производственные, фреймовые, формальные логические, на основе онтологий, семантические сети, древовидные, на основе множественного представления. Основными моделями обучаемого являются оверлейная и стереотипная. Используются следующие механизмы адаптации: авторский алгоритм, экспертная система, на основе производств, генетический алгоритм.

3. На сегодняшний день разработано множество механизмов адаптации, но эффективные механизмы персонализации процесса обучения требуют трудоёмкой работы по созданию курсов, так как основаны на системах производств.

4. Современные обучающие системы не учитывают специфику обучения проектной деятельности, в них отсутствует интеграция с профилем компетенций проектировщика, с его проектной деятельностью и базой проектных решений предприятия.

5. В обучающих системах можно выделить 3 уровня сложности разработки курса обучения: простая – полностью автоматическое построение курса, средняя – наполнение с помощью авторских редакторов, сложная – системы на основе правил, в которых необходимо разрабатывать производств для обеспечения адаптивной составляющей.

6. Методы формирования рекомендаций делятся на три больших класса: методы коллаборативной фильтрации, методы, анализирующие содержимое объектов, методы, основанные на знаниях. Авторский метод формирования рекомендаций на основе протокола проектных операций трехмерного моделирования деталей, выполняемых в САПР КОМПАС должен строиться на основе знаний, так как он позволяет расширять рекомендации и предлагать конкретные решения для пользователя.

7. В распространённых рекомендательных системах отсутствует оценка эффективности рекомендаций, последствий к которым они приведут. Также отсутствуют системы, подходящие для оценки проектировщика в процессе

проектной деятельности, формирования рекомендаций и корректировки профиля компетенций проектировщика.

8. Одним из подходов в подготовке проектировщиков для развития их прикладных компетенций является использование виртуальных тренажёров. Выделяются несколько основных подходов к проектированию виртуальных компонентов АОС: сценарный, на базе автоматного подхода, алгоритмический, гибридный. В настоящее время перспективным является гибридный подход, так как он сочетает сильные стороны автоматного и алгоритмического подходов. Активное использование виртуальной компоненты (виртуальные компьютерные тренажёры, виртуальная и дополненная реальности) является одной из современных парадигм построения АОС.

9. Основными методами для анализа действий проектировщиков при работе с виртуальными компонентами являются следующие: ручная проверка экспертом успеваемости обучаемых специалистов на основе логов, автоматическая пошаговая проверка процесса работы. Наиболее эффективным подходом к диагностике ошибок обучаемого при работе со сложными объектами является экспертная оценка. В основном такая оценка проводится на основе протокола действий или визуального наблюдения. Однако в случае большого количества обучаемых и ограниченного числа экспертов использовать данный метод не представляется возможным.

10. Определено место рекомендательной системы на основе протокола проектных операций в структуре автоматизированной обучающей системы между САПР КОМПАС и профилем проектировщика, что позволит оценить проектировщика в процессе проектной деятельности, сформировать рекомендации и скорректировать его профиль.

11. В соответствии с целью работы были выявлены следующие факторы, влияющие на эффективность деятельности проектировщика: неоптимальные действия, которые не изменяют состояние проектного решения (отменённые действия), неоптимальные одиночные действия над групповыми объектами, неоптимальные действия при построении специальных объектов (для которых

существуют специальные операции: фаска, скругление и т.д.), неоптимальные действия при переходах между операциями. Данные факторы должны лежать в основе математической модели системы формирования профиля компетенций проектировщика (СФПК).

12. Сформулированы основные гипотезы диссертационного исследования: если сформировать для обучаемого проектировщика персонализированный сценарий обучения на основе взаимодействия моделей пространства обучения (профиля проектировщика, теста, практического здания) с онтологической моделью предметной области машиностроительного проектирования, а также провести интеграцию с базой проектных решений предприятия, то это повысит скорость и качество обучения проектировщика; если в процессе работы проектировщика будут сформированы рекомендации на основе анализа выполненных им проектных операций трехмерного моделирования деталей, предлагающие заменить неоптимальные множества операций на множества операций с меньшим количеством действий, то это позволит повысить эффективность работы проектировщика.

Глава 2. Разработка математического обеспечения системы формирования профиля компетенций проектировщика

В данной главе предлагается схема системы формирования профиля компетенций проектировщика. Приводится описание основных подсистем и сценарии их взаимодействия.

Разрабатываются онтологическая модель предметной области машиностроительного проектирования, профиль проектировщика и метод формирования персонифицированного сценария обучения.

2.1. Общая организация системы формирования профиля компетенций проектировщика

Схема системы формирования профиля компетенций проектировщика (СФПК) представлена на рисунке 2.1.

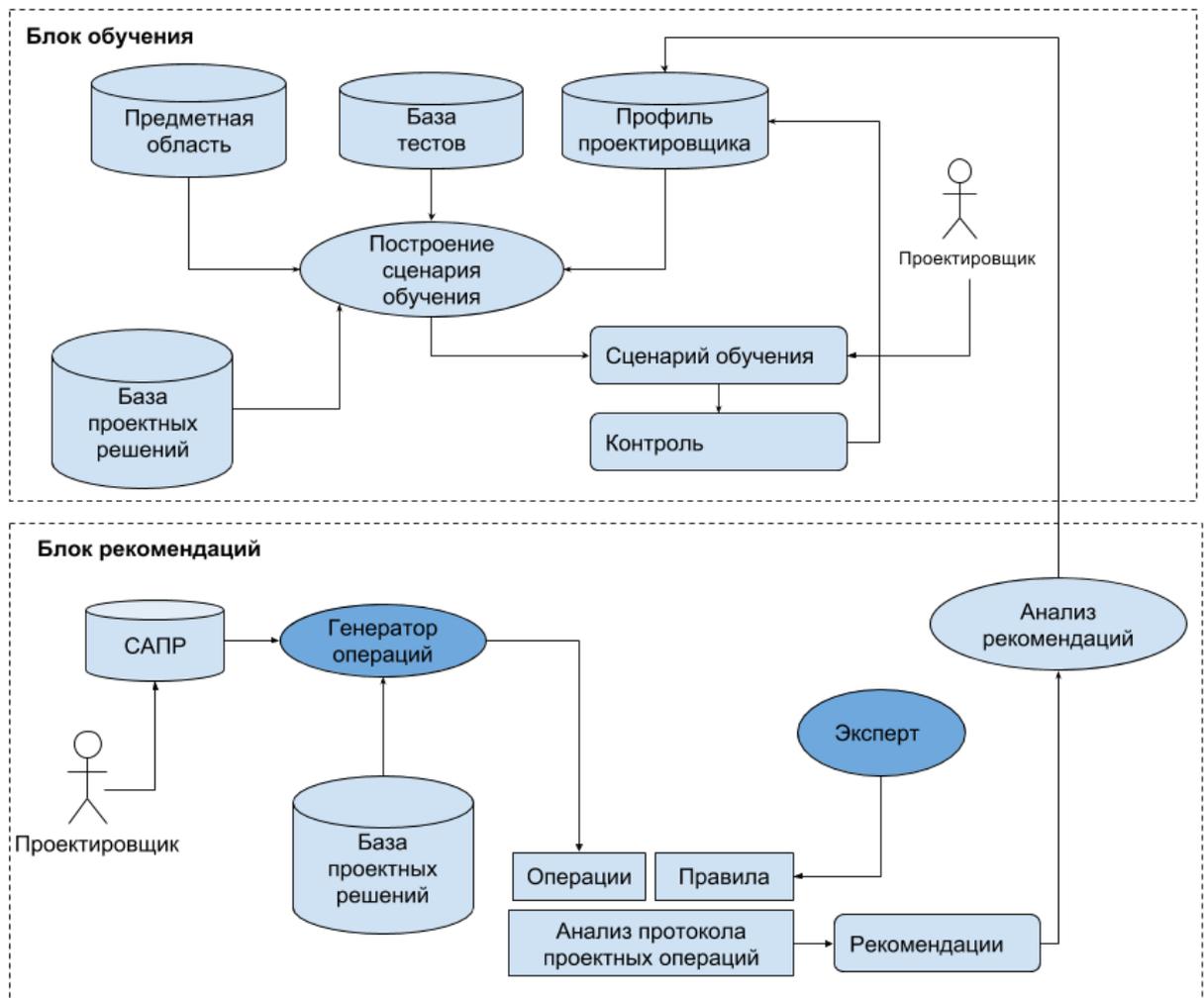


Рисунок 2.1 - Схема обучающей системы

В состав системы входят:

1. Блок обучения - отвечает за обучение проектировщиков (подбор материалов, тестирование, построение сценария обучения), состоит из следующих компонентов.

- Предметная область - хранит материал для обучения, множество учебных материалов и связей между ними.
- База тестов - тестовый материал для проверки уровня знаний.
- База проектных решений – практические задания для проверки навыков и умений проектировщика.
- Профиль проектировщика – хранит информацию об уровне знаний, результаты тестирования и т.д.
- Построение сценария обучения - построение сценария обучения и подбор тестов для обучаемого на основе анализа данных его профиля и информации из предметной области.
- Сценарий обучения - последовательность материалов для изучения проектировщиком и подборка тестовых заданий.
- Контроль – проверка тестовых и практических заданий, выполненных обучаемым проектировщиком.

2. Блок рекомендаций - отвечает за построение рекомендаций и диагностику уровня навыков и умений обучаемого проектировщика, состоит из следующих компонентов.

- САПР - Система автоматизированного проектирования, с которым работает проектировщик.
- Генератор операций - отслеживает действия обучаемого проектировщика и кодирует их для дальнейшего анализа.
- Генератор состояния - формирует состояние проекта на основе операций.
- Эксперт - эксперт по САПР, который заполняет базу правил для анализа проектных операций и формирования рекомендаций.

- Анализ протокола проектных операций - механизм формирования рекомендаций на основе предоставленных правил.
- Рекомендации - подобранные рекомендации для обучаемого проектировщика.
- Анализ рекомендаций - корректировка профиля проектировщика на основе сформированных рекомендаций.

2.2. Разработка процесса формирования профиля компетенций: режим сценария, режим поддержки проектировщика

В режиме сценария система строит персонифицированный сценарий обучения в соответствии с целями и успехами обучаемого проектировщика. Основными задействованными подсистемами являются: обучаемый, анализ проектов, виртуальный учитель, обработчик САПР, база проектных решений.

Обучаемый загружает проектные решения в базу проектных решений, устанавливает цели обучения, предоставляет свои проекты для анализа, изучает материал, проходит тестирование, выполняет практические задания и получает рекомендации.

Подсистема анализа проектов готовит проектные задания, проверяет проектные решения и отправляет результат виртуальному учителю.

Виртуальный учитель строит сценарий обучения, выдаёт тестовые и практические задания, обновляет профиль проектировщика.

Обработчик САПР следит за действиями проектировщика и отправляет информацию в подсистему анализа проекта.

База проектных решений предоставляет проектные задачи для обучаемого проектировщика.

Диаграмма последовательности представлена на рисунке 2.2.

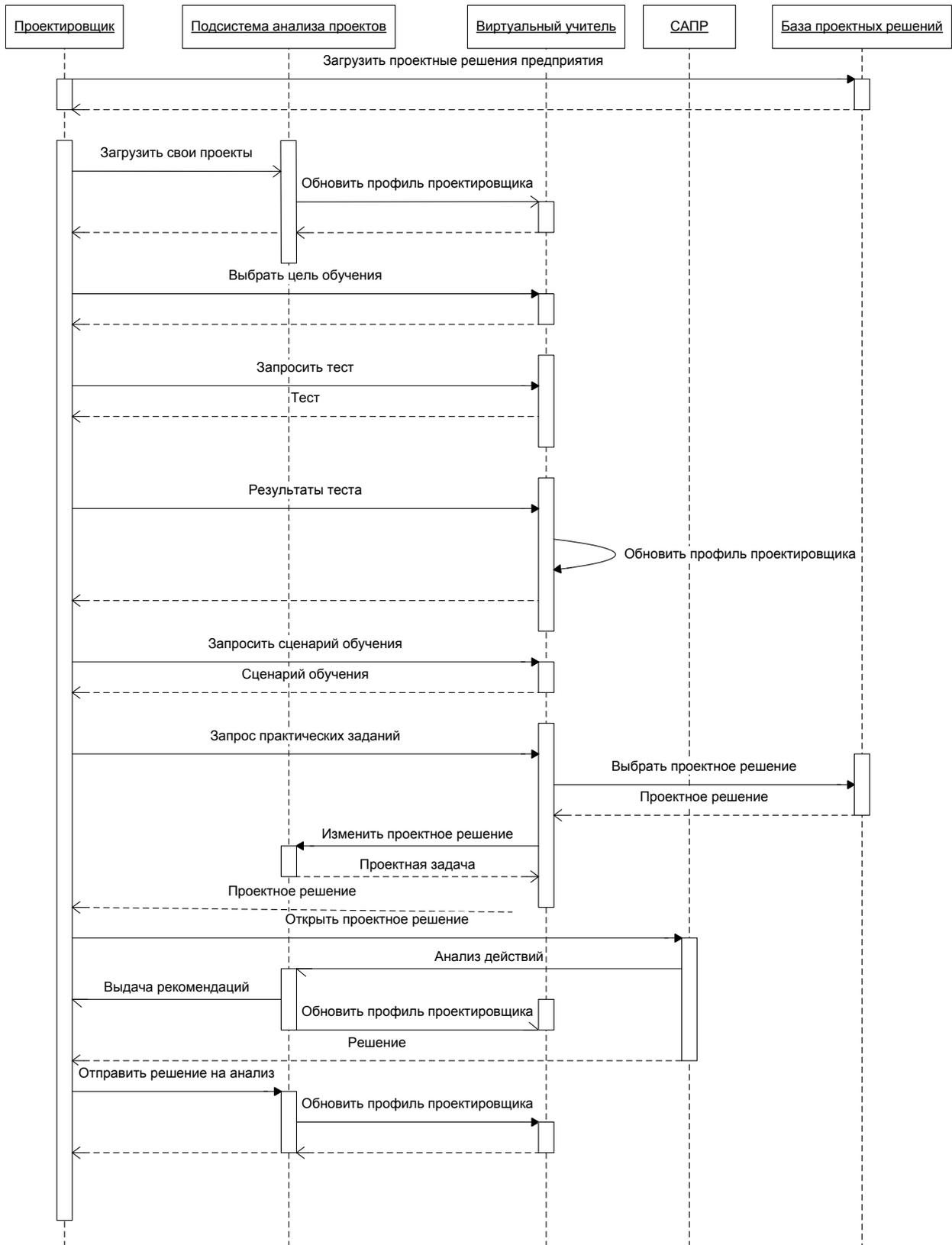


Рисунок 2.2 - Диаграмма последовательности режима сценария

Процесс обучения представлен ниже.

1. В систему загружается база проектных решений предприятия для отработки на них практических заданий.

2. Выбирается цель обучения.

3. Обучаемый проектировщик проходит первичное тестирование, на основе его результатов происходит формирование сценария обучения с использованием данных из баз предметной области, тестов и профиля проектировщика.

4. Проектировщик проходит обучение, состоящее как из теоретического материала, так и из тестовых заданий.

5. После теоретического изучения материала обучаемый проектировщик приступает к практическим заданиям.

6. В процессе работы с САПР генератор операций обрабатывает действия пользователя, на основе данной информации формируются протокол проектных операций.

7. На основе заложенных правил формируются рекомендации и предоставляются обучаемому проектировщику.

8. На основе задействованных операций, которые привели к выдаче рекомендаций, корректируется профиль проектировщика.

9. После изменения профиля проектировщика формируется новый сценарий обучения.

В режиме поддержки проектировщика обучаемый выполняет свои рабочие задания в САПР, параллельно с этим система строит рекомендации, и в случае необходимости предложит ему персонифицированный сценарий обучения. Основными задействованными подсистемами являются: обучаемый, анализ проектов, виртуальный учитель, обработчик САПР.

Диаграмма последовательности представлена на рисунке 2.3.

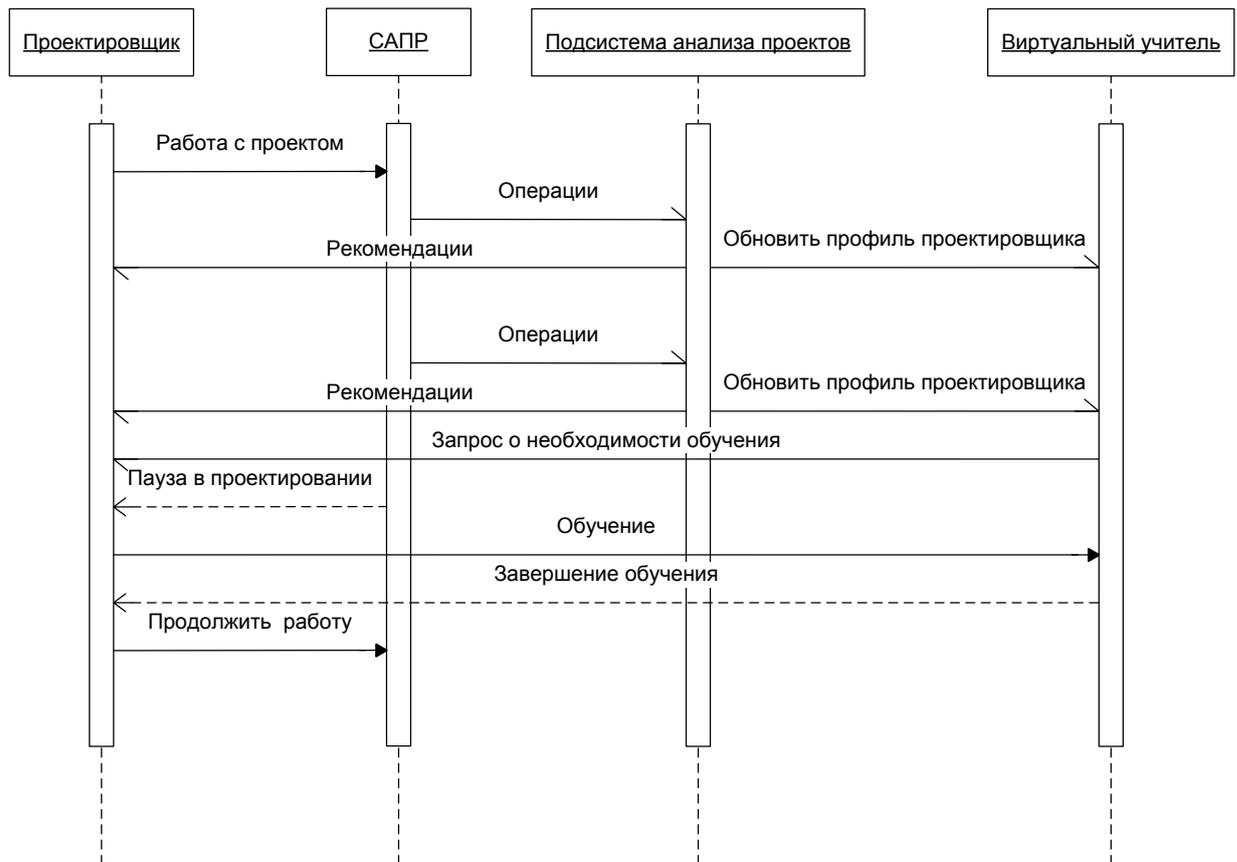


Рисунок 2.3 - Диаграмма последовательности режима поддержки проектировщика

Процесс обучения представлен ниже.

1. Проектировщик выполняет свои рабочие задания в САПР.
2. Обработчик САПР следит за действиями проектировщика и отправляет информацию в подсистему анализа проекта.
3. Подсистема анализа формирует рекомендации для проектировщика, а в случае необходимости предлагает ему персонализированный сценарий обучения.
4. Проектировщик проходит обучение.
5. Проектировщик продолжает выполнять свои рабочие задания.

2.3. Разработка моделей предметной области, профиля проектировщика и контроля

Модель предметной области

Разделим предметную область (ПрО) на три уровня: уровень схемы, представления и практический уровень [101].

Схема представляет множество знаний о ПрО – атомов, непересекающихся и далее неделимых для целей обучения ячеек знаний. Под знанием понимается совокупность формализованных сведений определенного объема, образующих целостное представление объекта проектирования на определенном уровне или описание определенного этапа проектной деятельности, предъявляемых субъекту в том или ином виде в ходе процесса обучения. Атомы могут группироваться по какому-либо принципу и образовывать понятия. Также атомы могут зависеть друг от друга, как знания, то есть для изучения атома необходимо сначала изучить атомы, зависящие от него. Если атом зависит от понятия, то необходимо изучить все атомы, входящие в это понятие.

Уровень представления – это множество учебных материалов (УМ) и справочников, которые представляют гипертекст [28, 71, 74, 127]. Каждый УМ покрывает один или несколько атомов или понятий. Покрытие понятия означает, что материал описывает все атомы, входящие в понятие. УМ характеризуется средней продолжительностью изучения, благодаря которому можно оценить время необходимое на проработку множества учебных материалов. Справочники являются дополнительным материалом, которые присутствуют как ссылки для необязательного ознакомления. Они прикрепляются к атомам или понятиям, при изучении которых появляются дополнительные ссылки к справочным материалам. Всего существует три типа вспомогательных материалов: справочники, определения и похожие УМ. Определения – это УМ, которые наиболее коротким образом описывают определяющий атом в понятии. Похожие УМ – это материалы, которые имеют наибольшее количество пересекающихся атомов знаний.

Практический уровень – это уровень для проверки навыков и умений обучаемого проектировщика и способности выполнять проектные действия на

основе приобретенных знаний автоматизированного проектирования и проектной практики. Проектное решение представляет собой множество узлов: объектов и операций. Объект – это часть проектных изделий, таких как отрезок, эскиз, деталь, сборка. Операция – это непосредственные действия над объектами, способ непосредственного создания или изменения объекта, например, вращение, перемещение и так далее. Каждой операции и объекту ставится в соответствие понятия, которые он описывает. Таким образом, связывается операция и соответствующий ему атом знания.

Типичная последовательность действий для построения изделия - это объект (один или несколько) → операция → новый объект (один или несколько), т.е. выбирается какой-то базовый объект, выполняется над ним операция и получается новый объект. При создании нового изделия, базовым объектом является одна из трёх плоскостей (XY, XZ, YZ). Для определения навыков и умений обучаемого для конкретного атома выбирается проектное решение и операция соответствующая данному атому. Обучаемый проектировщик должен получить базовый объект/объекты и выполнить над ним/ними требуемые операции, после чего система оценит правильность проектного решения, навыки и умения проектировщика.

Разделим учебный материал на непересекающиеся и далее неделимые для целей обучения ячейки знаний – атомы А. Соответственно каждый учебный материал D описывает множество таких атомов, $D_j = \{A_i\}$. Цель обучения определяет множество атомов, которые проектировщик должен изучить.

В предметной области определены следующие классы:

- атомы знаний;
- понятия – атомы, сгруппированные по какому-либо принципу;
- учебные материалы – материалы для изучения, одним из вариантов могут быть справочники - дополнительные материалы, которые присутствует как ссылки для необязательного ознакомления;
- цели обучения – группирует атомы, которые необходимы изучить,

• компетенции - группирует атомы, которые входят в данную компетенцию, является подклассом цели обучения.

Онтология предметной области [55, 45, 84, 125, 119, 114] имеет вид:

$$O = (PSL, T, R, F, Ax),$$

где $PSL = \{psl_i | i = 1..x\}$ – множество проектных решений,

T – термины прикладной области, которую описывает онтология.

Множество терминов представлено в виде:

$$T = \{C, In\},$$

где $C = \{A, P, D, GOAL, COMP\}$ – множество классов онтологии (A – атомы знаний, P – понятия, D – учебный материал, $GOAL$ – цель обучения, $COMP$ – компетенции), у класса «Учебный материал» определён слот «являетсяСправочником» с диапазоном значений *истина* или *ложь*,

In – множество объектов классов онтологии.

R – множество отношений между объектами онтологии:

$$R = \{R_{learn}, R_{part}, R_{next}\},$$

где R_{learn} – бинарное отношение «изучается_в», имеющее семантику «connected_to» и связывающее объекты классов онтологии («Атом», «Понятие») с объектами класса «УчебныйМатериал»,

R_{part} – бинарное отношение «состоит_в», имеющее семантику «part_of» и связывающее объекты классов онтологии («Атом», «Понятие») с объектами классов «Понятие», «ЦельОбучения»,

R_{next} – бинарное отношение «изучается_после», имеющее семантику «after_of» и связывающее объекты классов онтологии («Атом», «Понятие») с объектами класса «Понятие» и «Атом».

Множество интерпретирующих функций представлено в виде:

$$F = \{F_{atom_op}, F_{psl_a}, F_{edu}, F_{define}, F_{similar}, T\},$$

где $F_{atom_op} : A \rightarrow \{Operation\}$ – функция отображения объекта класса «Атом» на множество операций проектного решения,

$Fpsl_a : PSL \rightarrow \{A\}$ – функция отображения проектного решения на множество объектов класса «Атом»,

$Fedu : \{A\} \rightarrow \{D\}$ – функция построения упорядоченного множества учебных материалов для изучения определённых атомов знаний,

$Fdefine: P \rightarrow \{D\}$ – функция поиска учебных материалов, описывающих определённое понятие,

$Fsimilar: D \rightarrow \{D\}$ – функция поиска наиболее похожих учебных материалов,

$T: D \rightarrow \mathbb{Q}^+$ – дидактическая сложность материала.

Множество аксиом представлено в виде:

$$Ax = \{AxАНP, AxАНD, AxРАfP, AxРАfA, AxААfP\},$$

где $AxАНP$ – «атомы состоят в понятиях», если атом Y состоит в понятии X , которое состоит в понятии Z , то атом Y состоит в понятии Z , в виде SWRL [53, 8]:

$$\text{Понятие}(?x) \wedge \text{Атом}(?y) \wedge \text{Понятие}(?z) \wedge \text{состоит_в}(?y, ?x) \wedge \text{состоит_в}(?y, ?z) \rightarrow \text{состоит_в}(?y, ?z),$$

$AxАНL$ – «атомы состоят в целях обучения», если атом Y состоит в понятии X , которое состоит в цели обучения Z , то атом Y состоит в цели обучения Z , в виде SWRL:

$$\text{Понятие}(?x) \wedge \text{Атом}(?y) \wedge \text{ЦельОбучения}(?z) \wedge \text{состоит_в}(?y, ?x) \wedge \text{состоит_в}(?y, ?z) \rightarrow \text{состоит_в}(?y, ?z),$$

$AxАНD$ – «атомы состоят в учебных материалов», если атом Y состоит в понятии X , которое изучается в учебном материале Z , то атом Y изучается в учебном материале Z , в виде SWRL:

$$\text{Понятие}(?x) \wedge \text{Атом}(?y) \wedge \text{УчебныйМатериал}(?z) \wedge \text{состоит_в}(?y, ?x) \wedge \text{изучается_в}(?y, ?z) \rightarrow \text{изучается_в}(?y, ?z),$$

$AxРАfP$ – «атомы изучаются после атомов», если атом Y состоит в понятии X , которое изучается после понятия Z , а атом C состоит в Z , то атом Y изучается после C , в виде SWRL:

$$\text{Атом}(?y) \wedge \text{Понятие}(?x) \wedge \text{Понятие}(?z) \wedge \text{Атом}(?c) \wedge \text{состоит_в}(?y, ?x) \wedge \text{состоит_в}(?c, ?z) \wedge \text{изучается_после}(?x, ?z) \rightarrow \text{изучается_после}(?y, ?c),$$

$AxPAfA$ – «понятия изучаются после атомов», если атом Y состоит в понятии X , которое изучается после атома C , то атом Y изучается после C , в виде SWRL:

$$\text{Атом}(?y) \wedge \text{Понятие}(?x) \wedge \text{Атом}(?c) \wedge \text{состоит_в}(?y, ?x) \wedge \text{изучается_после}(?x, ?c) \rightarrow \text{изучается_после}(?y, ?c),$$

$AxAAfP$ – «атомы изучаются после понятий», если атом Y изучается после понятия X и атом C состоит в X , то атом Y изучается после C , в виде SWRL:

$$\text{Атом}(?y) \wedge \text{Понятие}(?x) \wedge \text{Атом}(?c) \wedge \text{состоит_в}(?c, ?x) \wedge \text{изучается_после}(?y, ?x) \rightarrow \text{изучается_после}(?y, ?c).$$

Рассмотрим пример модели предметной области (рисунок 2.4), примем следующие обозначения:

 Плоскость - атом,

 Эскиз (21) - понятие, (21) – количество атомов, состоящих в понятии,

 Урок №13. Вращение (7) - учебный материал, (7) – количество атомов, изучаемых в материале,

 История эскиза - справочный материал,

 - бинарное отношение «состоит_в»,

 - бинарное отношение «изучается_после»,

 - бинарное отношение «изучается_в».

Рассмотрим понятие «Построение дуги», оно состоит из 4 атомов: дуга по двум точкам; дуга по двум точкам и углу раствора; дуга по трём точкам; дуга, касательная к кривой. Пунктирной линией показана зависимость «Построения дуги» от атома «Определение», это значит, что нельзя изучать ни один атом в понятии «Построения дуги», пока не изучен атом «Определение».

Учебный материал «Что такое отрезок» покрывает 3 атома «Определение», «По двум точкам», «Длина и угол». При его изучении будет показана ссылка на справочный материал «Голландский угол», так как он ссылается на общий атом «Длина и угол».

Рассмотрим интеграцию ПрО с проектным решением (рисунок 2.5). Примем следующие обозначения:

Шайба – объект в САПР КОМПАС,

○ – операция в САПР КОМПАС,

→ – порядок построения.

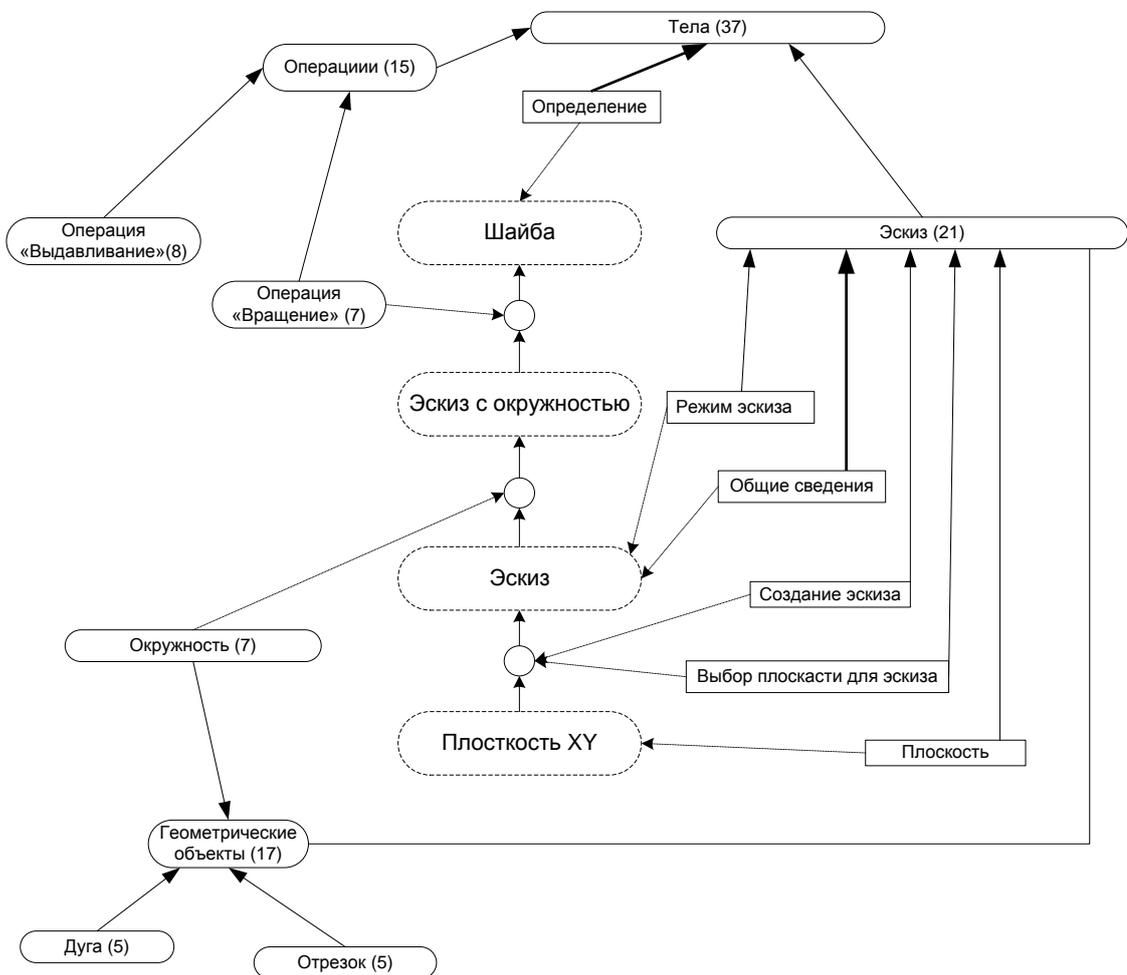


Рисунок 2.5 - Пример интеграции предметной области с деталью «Шайба»

Из рисунка видно, что объект «эскиз» связан с атомами «Режим эскиза» и «Общие сведения». А операция «Построение эскиза» с «Создание эскиза» и «Выбор плоскости для эскиза». Если при выполнении теста на построении шайбы

проектировщик совершит ошибку, то степени владения знаний и умений атомов, связанных с операциями построение эскиза, построение окружности и вращение уменьшатся.

Модель профиля проектировщика

Модель профиля проектировщика представляет собой оверлейную модель [25, 26, 27] и несёт информацию о степени владения знаниями и умениями, и навыками. Каждому атому ($A_i \in A$) сопоставлена степень владения знаниями ($p \in [0;1]$) обучаемым определённого атома и количество контрольных измерений данного атома знания ($c \in N$ – количество проверок), также каждому атому сопоставлена степень владения умениями ($p_{ab} \in [0;1]$) и количество контрольных измерений навыка ($c_{ab} \in N$ – количество проверок). Степени владения знаний, умений и навыков изменяются после контроля.

Модель профиля проектировщика имеет вид [66]:

$$U = (UA, P, C, P_{AB}, C_{AB}, ACT_G, ACT_C),$$

где $UA = \{a_i \in A | i = 1..nua\}$ – подмножество атомов знаний, которые необходимо изучить,

$P = A \rightarrow [0..1]$ – степень владения знаниями атома A ,

$C = A \rightarrow N$ – количество контрольных измерений знаний атома A ,

$P_{AB} = A \rightarrow [0..1]$ – степень владения умениями атома A ,

$C_{AB} = A \rightarrow N$ – количество контрольных измерений умений атома A ,

$P_S = A \rightarrow [0.. \infty]$ – степень владения навыками атома A ,

$C_S = A \rightarrow N$ – среднее время выполнения операций для атома A ,

$ACT_G = A \rightarrow N$ – суммарное количество действий при эталонном использовании навыка атома A ,

$ACT_C = A \rightarrow N$ – суммарное количество действий при текущем использовании навыка атома A .

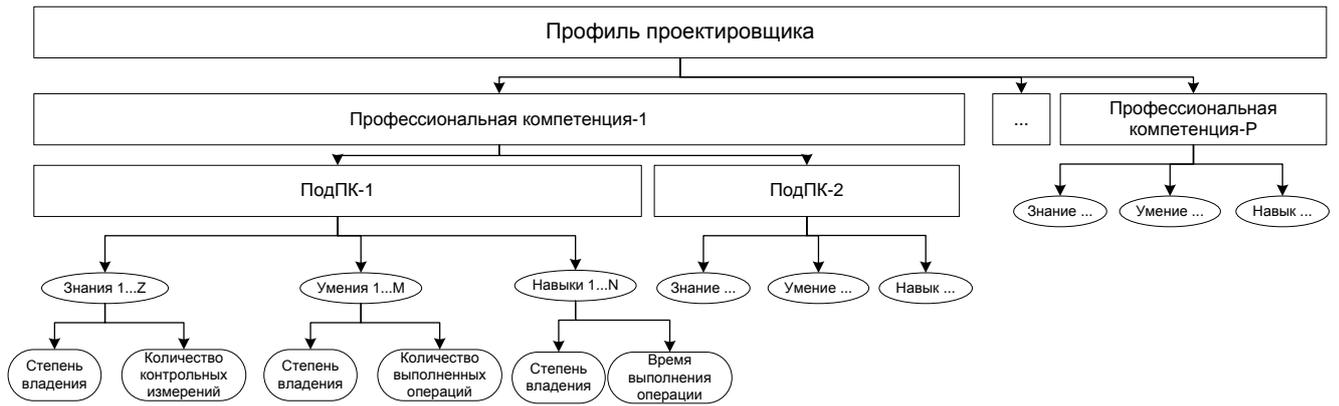


Рисунок 2.6 - Структура профиля проектировщика

На рисунке 2.6 показана структура профиля проектировщика с метриками для знаний, умений и навыков.

На рисунке 2.7 показан пример профиля проектировщика, и отмечены атомы знания, которые проектировщик уже изучил: «определение дуги»; «дуга, касательная к кривой»; «дуга по двум точкам и углу раствора»; «определение отрезка»; «перпендикулярный отрезок»; «определение окружности».

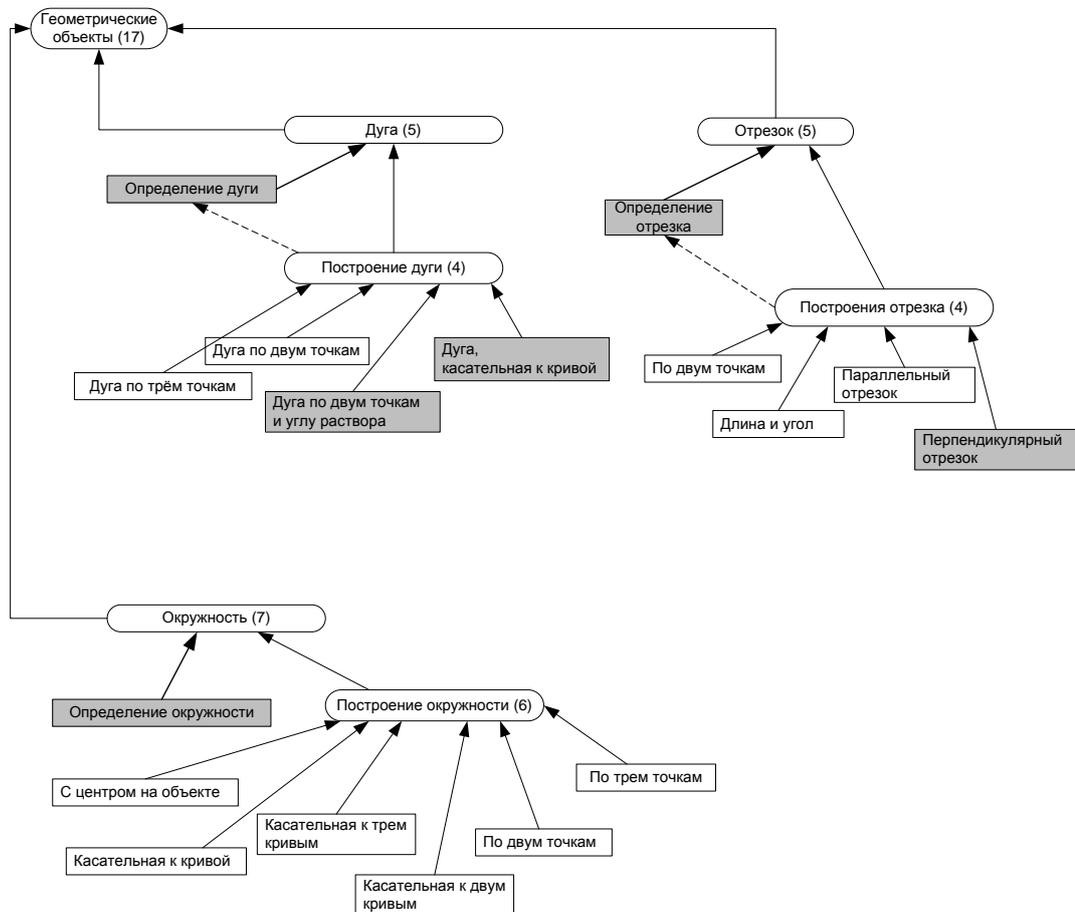


Рисунок 2.7 - Пример профиля проектировщика

Модель тестирования знаний

Модель тестирования знаний представляет тестовые задания с несколькими вариантами ответов, которые изменяют степень владения знаниями в профиле проектировщика. Каждому тестовому заданию ставится в соответствие множество атомов знаний, которые он оценивает. Правильный ответ изменяет степень владения знанием и количество контрольных измерений и по формуле:

$$c_{\text{после}} = c_{\text{до}} + 1,$$

$$p_{\text{после}} = \frac{(p_{\text{до}} * c_{\text{до}} + 1)}{c_{\text{до}} + 1},$$

неправильный:

$$c_{\text{после}} = c_{\text{до}} - 1,$$

$$p_{\text{после}} = \frac{(p_{\text{до}} * c_{\text{до}} - 1)}{c_{\text{до}} + 1}.$$

Модель теста имеет вид:

$$TZ = (q, ANS, TA, TAQ),$$

где Q = вопрос,

$ANS = \{ans_i | i = 1..n\}$ – множество ответов,

$TA = \{a_i \in A | i = 1..k\}$ - подмножество атомов знаний, которые оценивает тестовое задания,

$TAQ = \{q_i \in ANS | i = 1..m\}$ – множество правильных ответов.

Модель выполнения теста имеет вид:

$$UTZ = (U, TZ, QQ),$$

где U – профиль проектировщика,

TZ - тестовое задание,

$QQ = \{qq_i \in ANS | i = 1..k\}$ – подмножество ответов, выбранных обучаемым проектировщиком.

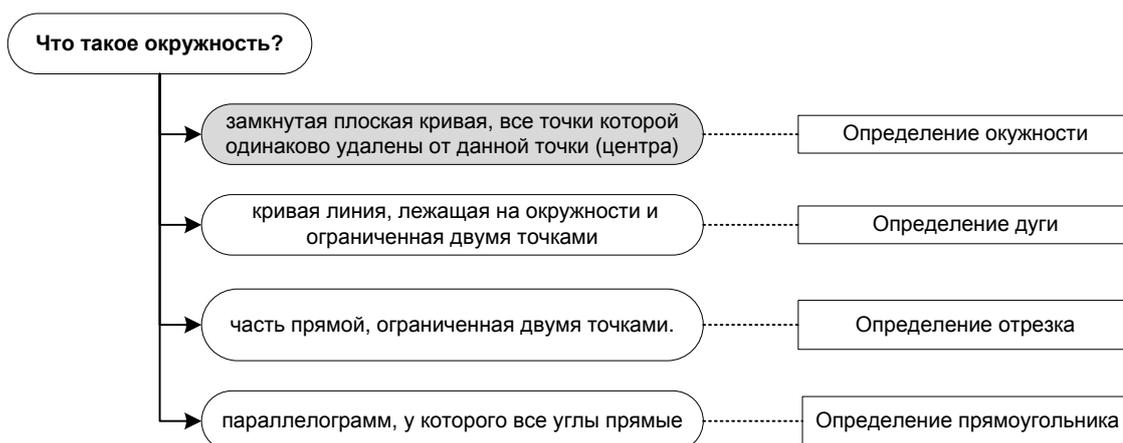


Рисунок 2.8 - Пример тестового задания

Пример тестового задания «Что такое окружность?» с 4 вариантами ответов представлен на рисунке 2.8. Каждый ответ связан с соответствующими атомами знаний. Правильный ответ «замкнутая плоская кривая, все точки которой одинаково удалены от данной точки (центра)» выделен на рисунке. Если обучаемый в процессе выполнения тестового задания даст правильный ответ, то степень владения знанием атома «Определение окружности» увеличится. А при выборе ответа «часть прямой, ограниченная двумя точками», степень владения знаниями атомов «Определение окружности» и «Определение отрезка» уменьшится.

Модель проектного задания

Для проверки умений и навыков проектировщика используются практические задания по разработке проектного решения. Система автоматически сравнивает решение с эталоном и корректирует степени владения навыками, умениями, количество контрольных измерений конкретных атомов знаний и время выполнения операций. Анализ проектных решений в виде диаграмматических электронных моделей производится на основе [150].

Модель проектного задания имеет вид:

$$PTZ = (pq, std_start, std_finish),$$

где pq – описание задания,

$std_start \in PSL$ – начальное проектное решение,

$std_finish \in PSL$ – эталонное проектное решение.

Пример проектного задания «Построить кольцо с внешним диаметром 50, внутренним диаметром 30 и высотой 10. На внутренней стороне должна быть фаска с длиной 1 и углом 45° , на внешней – с длиной 2», представлен на рисунке 2.9.

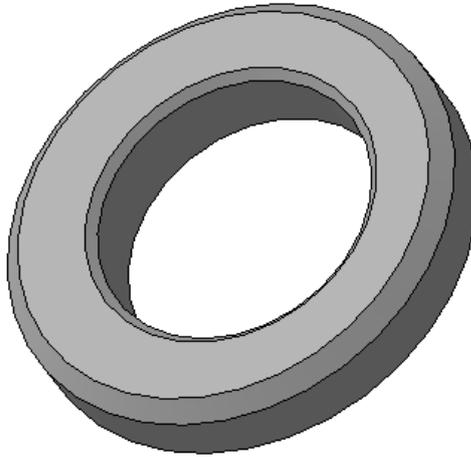


Рисунок 2.9 - Пример проектного задания «Кольцо»

Интеграция данного проектного задания с предметной областью показана на рисунке 2.10.

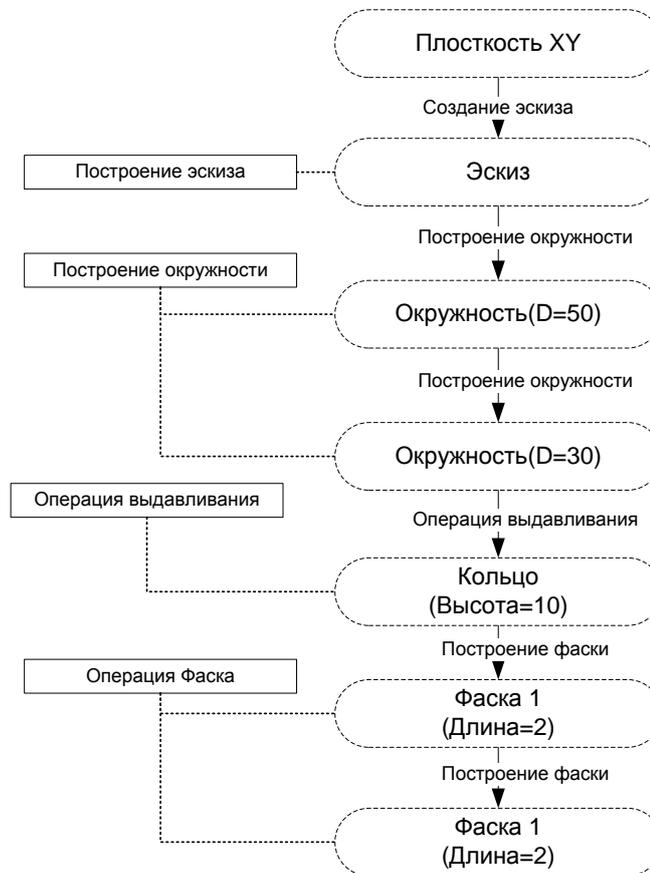


Рисунок 2.10 - Интеграция умений и навыков с проектным заданием

Модель решения проектного задания имеет вид:

$$UPTZ = (U, PTZ, psl, Ap, An, T),$$

где U – профиль проектировщика,

PTZ – практическое задание,

$psl \in PSL$ – проектное решение,

$Ap = \{a_i \in A | i = 1..k\}$ - подмножество атомов знаний, для которых подтвердились умения обучаемого проектировщика,

$An = \{a_i \in A | i = 1..n\}$ - подмножество атомов знаний, для которых не подтвердились умения обучаемого проектировщика,

$T_S = A \rightarrow N$ – время выполнения операций для атома A ,

$ETS = A \rightarrow N$ – эталонное время выполнения операций для атома A .

После анализа проектного решения, корректируется профиль проектировщика. Множество атомов знаний, для которых подтвердились умения обучаемого, изменяют степени владения навыками и умениями, количество контрольных измерений и среднее время выполнения операций по формуле:

$$\begin{aligned} c_{ab_{\text{после}}} &= c_{ab_{\text{до}}} + 1, \\ p_{ab_{\text{после}}} &= \frac{(p_{ab_{\text{до}}} * c_{ab_{\text{до}}} + 1)}{c_{ab_{\text{до}}} + 1}, \\ p_{s_{\text{после}}} &= \frac{(p_{s_{\text{до}}} * c_{ab_{\text{до}}} + \frac{ETS}{T_S})}{c_{ab_{\text{до}}} + 1}, \\ c_{s_{\text{после}}} &= \frac{c_{s_{\text{до}}} * c_{ab_{\text{до}}} + t_s}{c_{ab_{\text{до}}} + 1}, \end{aligned}$$

для которых не подтвердились умения обучаемого:

$$\begin{aligned} c_{ab_{\text{после}}} &= c_{ab_{\text{до}}} - 1, \\ p_{ab_{\text{после}}} &= \frac{(p_{ab_{\text{до}}} * c_{ab_{\text{до}}} - 1)}{c_{ab_{\text{до}}} - 1}. \end{aligned}$$

На рисунке 2.11 показан механизм сравнения эталонного решения с решением обучаемого проектировщика. Прерывистыми линиями между двумя проектами показаны некорректно выполненные операции или невыполненные вообще. Стрелками показаны изменения умений в профиле проектировщика, для корректно выполненных операций степень владения соответствующими умениями выросла

(построение эскиза, построение окружности, операция выдавливания, операция фаска), для некорректно выполненных уменьшилась (построение окружности, операция фаска). Умения, для которых произошло как уменьшение, так и увеличение степени владения, суммарно остались неизменными (построение окружности, операция фаска).

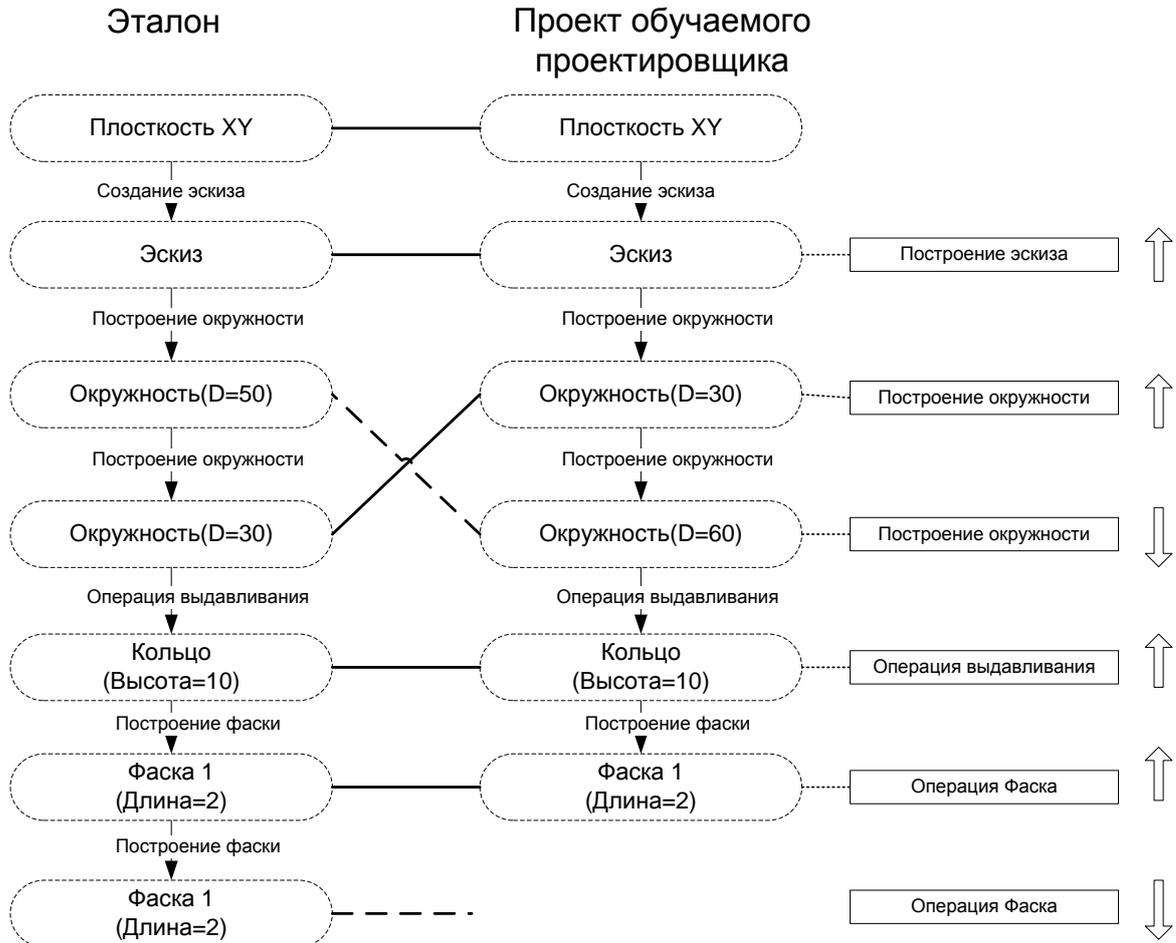


Рисунок 2.11 - Анализ проектного решения обучаемого проектировщика

Модель сценария

Сценарий состоит из этапов, содержащих множество учебных материалов, тестовых и проектных упражнений. Для каждого сценария определяется подмножество атомов знаний, которое обязан усвоить обучаемый проектировщик.

Модель сценария имеет вид:

$$ST = (A, P, D, STZ, SPTZ, PERIOD, Fd_d, Fphase_tz, Fphase_ptz, Fphase_d, Fphase_first, Fnext_phase),$$

где $A = \{a_i | i = 1..n\}$ – множество атомов знаний,

$P = \{p_i | i = 1..j\}$ – множество понятий,

$D = \{d_i | i = 1..k\}$ – множество учебных материалов,

$STZ = \{tz_i | i = 1..m\}$ – множество тестовых заданий,

$SPTZ = \{ptz_i | i = 1..l\}$ – множество практических заданий,

$PHASE = \{pz_i | i = 1..h\}$ – множество этапов сценария,

$Fd_d = D \rightarrow D$ – функция следования учебных материалов,

$Fphase_d = PHASE \rightarrow D^*$ – функция принадлежности подмножеству учебных материалов определённому этапу сценария,

$Fphase_tz = PHASE \rightarrow TZ^*$ – функция принадлежности подмножества тестовых заданий определённому этапу сценария,

$Fphase_ptz = PHASE \rightarrow PTZ$ – функция принадлежности практического задания определённому этапу сценария,

$Fnext_phase = PHASE \rightarrow PHASE$ – функция следования этапов сценария,

$Fphase_first = PHASE \rightarrow D$ – функция первого учебного материала в определённом этапе сценария.

2.4. Разработка метода формирования персонафицированного сценария обучения

Алгоритм состоит из 10 шагов и представлен ниже [97].

Шаг 1. Выбор цели обучения.

Первым шагом выбирается цель обучения $goal \in GOAL$. Каждой цели обучения соответствует множество атомов знаний, которые необходимо изучить проектировщику. Профиль проектировщика заполняется атомами исходя из уровня обучения:

$$U[UA] = \{a \in A | R_{part}(a, goal)\}.$$

При инициализации профиля степень владения знаниями, умениями и навыками устанавливается в 0, количество контрольных измерений также в 0:

$$\forall a \in U[UA]: U[P(a)] = 0,$$

$$\forall a \in U[UA]: U[C(a)] = 0,$$

$$\forall a \in U[UA]: U[P_S(a)] = 0,$$

$$\begin{aligned}\forall a \in U[UA]: U[C_S(a)] &= 0, \\ \forall a \in U[UA]: U.P[P_{AB}(a)] &= 0, \\ \forall a \in U[UA]: U.C[C_{AB}(a)] &= 0.\end{aligned}$$

Шаг 2. Загрузка базового сценария обучения.

Вторым шагом выбирается базовый сценарий обучения и загружается первый этап обучения выбранного сценария ($phase \in PHASE$).

Шаг 3. Изучение материала.

Третьим шагом обучаемый проектировщик проходит заданный этап сценария. При выводе учебного материала ($d \in D$) также будут выдаваться ссылки на дополнительные материалы.

Справочники. Из предметной области выбираются справочные материалы, которые связаны с атомами знаний выбранного учебного материала, правило для выбора имеет вид:

$$\{h \in D: \text{являетсяСправочником}(h) \wedge |\{a \in A: R_{learn}(a, h) \wedge R_{learn}(a, d)\}| > 0\}.$$

Определения. Из предметной области выбираются учебные материалы, которые описывают только один атом знаний, правило для выбора имеет вид:

$$\{h \in D: |\{a \in A: R_{learn}(a, h)\}| = 1 \wedge |\{a \in A: R_{learn}(a, h) \wedge R_{learn}(a, d)\}| > 0\}.$$

Похожие материалы. Из предметной области выбираются учебные материалы, которые имеют наибольшее количество общих атомов знаний (больше порога t). Правило для выбора имеет вид:

$$\{h \in D: |\{a \in A: R_{learn}(a, h) \wedge R_{learn}(a, d)\}| \geq t\}.$$

Шаг 4. Выполнение тестовых заданий.

Четвёртый шагом обучаемому проектировщику выдаётся подмножество тестовых заданий. Для каждого тестового задания обучаемый формирует подмножество ответов. В соответствии с ответами модифицируются степени владения знаниями и количество контрольных измерений в профиле проектировщика. Правильные ответы увеличивают степень владения знаниями, неправильные – уменьшают.

Шаг 5. Выполнение проектных заданий.

Пятым шагом выбираются практические задания и выдаются обучаемому проектировщику (U). В процессе выполнения каждого задания формируется проектное решение $psl2 \in PSL$, которое проверяется на корректность с эталоном $psle \in PSL$. Если решение правильное, то степени владения умениями соответствующих атомов увеличиваются, в противном – уменьшаются. Также изменяются степени владения навыками и среднее время выполнения операций.

Шаг 6. Анализ уровня знаний, умений и навыков проектировщика.

Если после прохождения этапа сценария обучения, контрольного теста и практического теста степени владения знаниями, умениями и навыками считаются удовлетворительными, то обучаемый переходит на следующий этап обучения на четвёртый шаг, если все этапы сценария пройдены, то обучаемый заканчивает обучение. В противном случае выполняется запрос на генерацию персонафицированного сценария обучения.

Шаг 7. Выбор атомов знаний, требующих дополнительного изучения.

Седьмым шагом начинается генерация персонафицированного сценария обучения. Сначала выбираются атомы, которые изучались на данном этапе, $A1 = \{a \in A : \exists d \in Fphase_d(phase) \wedge R_{learn}(a, d)\}$. Далее необходимо оставить только те атомы знаний, которые обучаемый проектировщик не знает, степень владения которых ниже порога thr :

$$A2 = \{a \in A1 \mid U[p(a)] < thr\},$$

$$B2 = \{a \in A1 \mid U[P_{AB}(a)] < thr\}.$$

Шаг 8. Подбор учебных материалов.

Теперь необходимо выбрать учебные материалы, которые покрывают данные атомы:

$$D1 = \{d \in D \mid \exists a \in A2 \wedge R_{learn}(a, d)\}.$$

В целях сокращения времени обучения выберем материалы, которые имеют наименьшее предполагаемое время изучения. То есть необходимо сформировать множество учебных материалов $D2$, в которых изучаются атомы из множества $A2$

и суммарное предполагаемое время изучения было бы минимально, также необходимо минимизировать количество повторов с уже пройденным материалом DL:

$$\left\{ \begin{array}{l} (\forall a \in A2, \exists d \in D2) R_{learn}(a, d) \\ \sum_{i=0}^{|D2|} T(d_i) \rightarrow \min \\ |D1 \cap DL| \rightarrow \min. \end{array} \right.$$

Необходимо упорядочить данное множество в соответствии с зависимостями атомов.

Шаг 9. Формирование множества тестовых и проектных заданий.

Девятым шагом формируется минимальное множество тестовых заданий покрывающих атомы знаний A2:

$$TZA = \{t \in STZ \mid \exists a \in A2 \wedge a \in t[A]\}.$$

Также формируется множество практических заданий, выбирается практическое задание $ptz \in SPTZ$, такое что:

$$\left\{ \begin{array}{l} |B2 \cap Fpsl_a(ptz[standart_finish])| \rightarrow \max \\ |B2 \Delta Fpsl_a(ptz[standart_finish])| \rightarrow \min \end{array} \right.$$

Шаг 10. Контрольное тестирование.

После прохождения персонифицированного сценария обучения проводится контрольное тестирование, если и в этом случае характеристики обучаемого не соответствуют ожидаемым, то строится повторный сценарий, отличием которого является, что множество учебных материалов D2 подбирается с наименьшим повторением относительного предыдущего сценария.

На рисунке 2.12 показана IDEF0-диаграмма взаимодействия моделей пространства обучения.

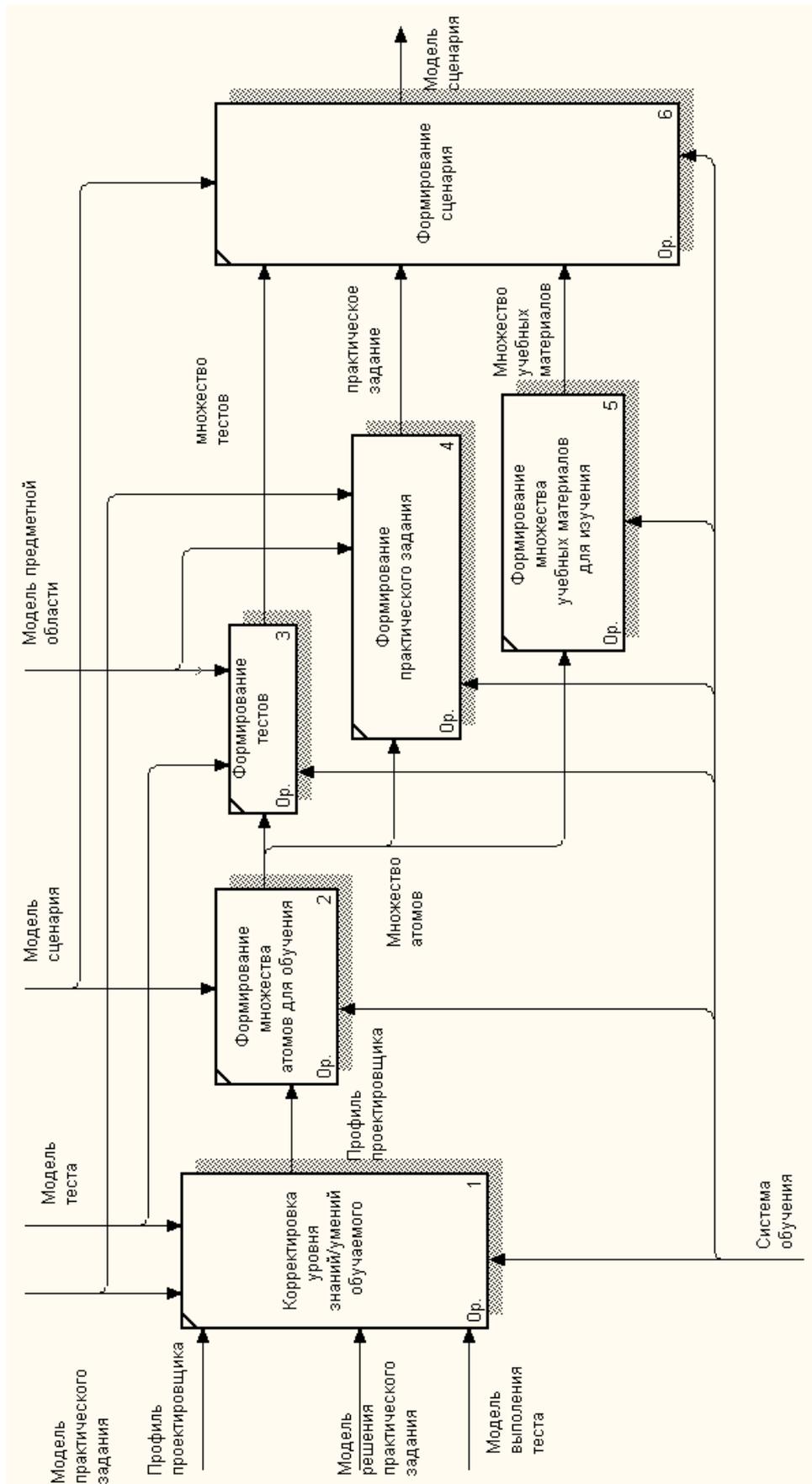


Рисунок 2.12 - Взаимодействие моделей пространства обучения [98]

2.5. Выводы и рекомендации

1. Разработана система формирования профиля компетенций проектировщика, состоящая из двух основных подсистем: блок обучения (компоненты: предметная область, база тестов, база проектных решений, профиль проектировщика, построение сценария обучения, сценарий обучения, контроль) и блок рекомендаций (компоненты: САПР, генератор операций, генератор состояния, эксперт, анализ протокола проектных операций, рекомендации, анализ рекомендаций).

2. Предложены два режима формирования профиля компетенций: режим сценария и режим поддержки проектировщика. В режиме сценария система строит адаптивный сценарий обучения в соответствии с целями и успехами обучаемого проектировщика. В режиме поддержки проектировщика система отслеживает его действия, выполняемые в САПР КОМПАС, формирует рекомендации, в случае необходимости строит персонафицированный сценарий обучения.

3. Разработана онтологическая модель предметной области, отличающаяся введением классов, атомов и функций, ориентированных на предметную область автоматизированного машиностроительного проектирования, и позволяющая персонафицировать процесс обучения. Предметная область (ПрО) содержит три уровня: уровень схемы, представления и практический уровень. Схема представляет множество знаний о ПрО – атомов, непересекающихся и далее неделимых для целей обучения ячеек знаний. Уровень представления – это множество учебных материалов (УМ) и справочников, которые представляют гипертекст. Практический уровень – это уровень для проверки умений и навыков проектировщика, способности выполнять проектные действия на основе приобретенных знаний автоматизированного проектирования и проектной практики, представлен множеством проектных решений.

4. Проработана интеграция предметной области с базой проектных решений через соответствие атомов знаний ПрО с проектными операциями.

5. Профиль проектировщика построен на основе оверлейной модели и представляет знания, умения и навыки, как дополнительные характеристики

атомов предметной области (степень владения знаниями атома, количество контрольных измерений знаний атома, степень владения умениями атома, количество контрольных измерений умений атома, степень владения навыками атома, среднее время выполнения операций для атома, суммарное количество действий при эталонном использовании навыка атома, суммарное количество действий при текущем использовании навыка атома).

6. Разработана модель тестового задания в виде вопроса с несколькими вариантами ответов. Отличительной особенностью является интеграция ответов с атомами знаний предметной области, позволяющая корректировать степень владения знаниями в профиле проектировщика.

7. Разработаны модель проектного задания, состоящая из начального проектного и эталонного решений, и алгоритм анализа проектного решения, позволяющий корректировать степени владения умениями и навыками в профиле проектировщика.

8. Разработан метод формирования персонифицированного сценария обучения, отличающийся использованием динамических механизмов взаимодействия моделей пространства обучения (профиля проектировщика, теста, практического задания) с онтологической моделью предметной области и обеспечивающий формирование персонифицированных компетенций проектировщика.

9. Рекомендуются использование разработанных моделей пространства обучения и метода формирования персонифицированного сценария обучения для минимизации суммарного времени обучения, с учётом практической подготовки обучаемого, лёгкости наполнения предметной области, наличия материалов для дополнительного изучения.

Глава 3. Разработка системы рекомендаций и виртуальной компоненты для формирования профиля компетенций проектировщика

В данной главе рассматриваются вопросы разработки рекомендаций для формирования профиля компетенций проектировщика: построение списка рекомендаций, метод формирования рекомендаций и оценка их эффективности.

Также разработана модель виртуального компонента автоматизированной обучающей системы и метод автоматической диагностики ошибок в тренажёрах на основе автоматной модели.

3.1. Формальное описание задачи

Расчёт сокращения действий выполняется по следующей формуле:

$$R(U, TZ) = \frac{SD(U, TZ) - SD(U^*, TZ)}{SD(U, TZ)},$$

где U – профиль проектировщика до обучения,

U^* – профиль проектировщика после обучения,

TZ – проектное задание,

$SD(U, TZ)$ – количество выполненных действий проектировщиком с профилем U для решения задания TZ .

Для выполнения проектного задания проектировщик формирует проектное решение P :

$$P = FP(U, TZ),$$

где FP – процесс формирования проектного решения.

Количество выполненных действий для решения проектного задания:

$$SD(u, tz) = SP(P) = \sum_{i=0}^n [D(Op_i) + DD(Op_i, Op_{i+1})],$$

где $SP(P)$ – количество действий в проектном решении P ,

$D(Op_i)$ – количество выполненных действий для построения операции $Op_i \in P, i = 0 \dots n, n$ – количество операций в проектном решении P ,

$DD(Op_i, Op_{i+1})$ – количество выполненных действий для перехода от операции Op_i к операции Op_{i+1} .

Во введенных обозначениях общее правило функционирования системы формирования рекомендаций (СФР) можно представить в следующем виде. СФР, получая на входе информацию о профиле проектировщика u , на основе анализа его проектных действий выдает на выходе управление (рекомендацию):

$$X = A(U),$$

где A – алгоритм управления (процессом обучения).

Сформированная рекомендация переводит проектировщика из текущего состояния в состояние U^* , такое что:

$$SD(U^*, TZ) < SD(U, TZ).$$

Положим, что модель профиля проектировщика, связывающая его наблюдаемые входы и выходы, имеет вид:

$$U' = M(U, X),$$

где M – функция изменения профиля проектировщика U под воздействием рекомендаций X .

Задачу синтеза оптимального управления X можно записать в виде:

$$SP(FP(M(U, X), TZ)) \rightarrow \min.$$

$F_{state}, F_{group}, F_{special}, F_{trans}$ – факторы, влияющие на эффективность деятельности проектировщика. Для количественной оценки факторов введём функции $V_{state}, V_{group}, V_{special}, V_{trans}$ значения которых позволят численно оценить значения факторов, где $V_{state} \in [0; 1]$ – доля действий которые не изменяют состояние проектного решения (отменённые действия),

$$V_{state}(U) = \sum_{i=0}^{|Ast|} (1 - \frac{U[ACT_G(Ast_i)]}{U[ACT_C(Ast_i)]}),$$

где Ast – множество умений для уменьшения действий, которые не влияют на проектное решение,

$V_{group} \in [0; 1]$ – доля лишних одиночных действий над групповыми объектами по сравнению с максимально возможным,

$$V_{special}(U) = \sum_{i=0}^{|Ag|} (1 - \frac{U[ACT_G(Ag_i)]}{U[ACT_C(Ag_i)]}),$$

где Ag – множество умений для работы с групповыми объектами,

$V_{special} \in [0; 1]$ – доля лишних действий для построения специальных объектов (для которых существуют специальные операции: фаска, скругление и т.д.) по сравнению общим количеством действий для построения специальных объектов,

$$V_{special}(U) = \sum_{i=0}^{|As|} \left(1 - \frac{U[ACT_G(As_i)]}{U[ACT_C(As_i)]}\right),$$

где As – множество умений для построения специальных объектов,

$V_{trans} \in [0; 1]$ – доля лишних действий при переходах между операциями по сравнению с общим количеством действий при переходах,

$$V_{trans}(U) = \sum_{i=0}^{|At|} \left(1 - \frac{U[ACT_G(At_i)]}{U[ACT_C(At_i)]}\right),$$

где At – множество умений для оптимальных переходов между операциями.

Проектное решение P характеризуется следующими параметрами:

$$P = P(SZ_{all}, SZ_{group}, SZ_{special}, SZ_{trans}),$$

где SZ_{all} – количество действий в проектном решении,

SZ_{group} – количество действий над групповыми объектами,

$SZ_{special}$ – количество действий для построения специальных объектов,

SZ_{trans} – количество действий для перехода между операциями.

Допустим P^* является наиболее оптимальным решением (критерий: наименьшее количество действий) проектного задания TZ , тогда количество действий, требуемое для его выполнения проектировщиком с профилем U :

$$SD(U, TZ) = SZ_{all}(P^*) + SZ_{all}(P^*) * \frac{V_{state}(U)}{1-V_{state}(U)} + SZ_{group}(P^*) \frac{V_{group}(U)}{1-V_{group}(U)} +$$

$$SZ_{special}(P^*) \frac{V_{special}(U)}{1-V_{special}(U)} + SZ_{trans}(P^*) \frac{V_{trans}(U)}{1-V_{trans}(U)}.$$

Таким образом, задача СФР - синтез оптимальных рекомендаций X с целью минимизации параметров профиля проектировщика:

$$V_{state} \rightarrow 0, V_{group} \rightarrow 0, V_{special} \rightarrow 0, V_{trans} \rightarrow 0.$$

3.2. Формирование списка рекомендаций

Для формирования списка рекомендаций использовался метод интервьюирования экспертов предметной области и анализ документа «КОМПАС-

3D V16. Руководство пользователя» [108]. Анализ документа выявил 1957 потенциальных рекомендаций. Ниже представлена часть полученного списка.

Рекомендации описаны по схеме:

- текст рекомендации,
- область действия (на какие операции распространяется),
- условие для формирования рекомендации,
- объяснение за счёт чего достигается уменьшение количества выполняемых действий,
- количественная оценка уменьшения количества выполняемых действий.

Не следует задавать параметр «Направление первой стороны» для операции фаски, если её стороны равны, так как результат построения не будет зависеть от данного параметра.

Область действия: операция «Фаска».

Условие: существует операция «Фаска», у которой значением параметра «Направление первой стороны» является «Вторая сторона».

Причина: целенаправленное указание параметра «Направление первой стороны» для операции «Фаска» при равенстве сторон не имеет смысла, поэтому данное действие является лишним.

Результат: уменьшение количества действий на 1 или в среднем на 17%.

При построении фаски для множества рёбер с одинаковыми параметрами операции, постарайтесь выбрать как можно большее количество ребер.

Область действия: операция «Фаска».

Условие: существуют операции «Фаска» с одинаковыми параметрами.

Причина: выполнение операции «Фаска» для множества рёбер уменьшает количество получаемых в результате объектов, что повышает производительность САПР, также уменьшает количество необходимых действий по сравнению с выполнением операций над каждым ребром отдельно.

Результат: в среднем для построения фаски для одного ребра необходимо 7 действий (выбрать операцию, выбрать ребро, задать 4 параметра, выполнить). При выполнении операции над множеством рёбер необходимо выполнить $6 + X$ действий, где X - количество рёбер. Следовательно, уменьшение количества действий составит:

$$\text{абсолютное значение} = (3+p) * e - (2+p+e),$$

$$\text{относительное значение} = \frac{2+p+e}{(3+p)*e},$$

где p – количество заданных параметров; e – количество рёбер.

Конкретные значения для 4 заданных параметров представлены в таблице 3.1.

Таблица 3.1 - Уменьшение количества действий для рекомендации «Не стройте фаску для каждого ребра в отдельности»

Количество рёбер	Количество действий без использования рекомендации	Количество действий с использованием рекомендации	Абсолютное значение уменьшения количества действий	Относительное значение уменьшения количества действий, %
2	14	8	6	43
3	21	9	12	57
4	28	10	18	64
5	35	11	24	69

При построении скругления для множества рёбер с одинаковыми параметрами операции, постарайтесь выбрать как можно большее количество рёбер.

Область действия: операция «Скругление».

Условие: существуют операции «Скругление» с одинаковыми параметрами.

Причина: выполнение операции «Скругление» для множества рёбер уменьшает количество получаемых в результате объектов, что повышает производительность САПР, также уменьшает количество необходимых действий по сравнению с выполнением операцией над каждым ребром отдельно.

Результат: в среднем для построения скругления для одного ребра необходимо 6 действий (выбрать операцию, выбрать ребро, задать 3 параметра, выполнить). При выполнении операции над множеством рёбер необходимо выполнить $5 + X$ действий, где X - количество рёбер. Следовательно, уменьшение количества действий составит:

$$\text{абсолютное значение} = (2+p) * e - (1+p+e),$$

$$\text{относительное значение} = \frac{1+p+e}{(2+p)*e},$$

где p – количество заданных параметров; e – количество рёбер.

Конкретные значения для 3 заданных параметров представлены в таблице 3.2.

Таблица 3.2 - Уменьшение количества действий для рекомендации «Не стройте скругление для каждого ребра в отдельности»

Количество рёбер	Количество действий без использования рекомендации	Количество действий с использованием рекомендации	Абсолютное значение уменьшения количества действий	Относительное значение уменьшения количества действий, %
2	12	7	5	42
3	18	8	10	56
4	24	9	15	63
5	30	10	20	67

При построении операции «Уклон» для множества граней с одинаковыми параметрами операции, постарайтесь выбрать как можно большее количество граней.

Область действия: операция «Уклон».

Условие: существуют операции «Уклон» с одинаковыми параметрами.

Причина: выполнение операции «Уклон» для множества граней уменьшает количество получаемых в результате объектов, что повышает производительность САПР, также уменьшает количество необходимых действий по сравнению с выполнением операцией над каждым ребром отдельно.

Результат: в среднем для построения уклона для одной грани необходимо 5 действий (выбрать операцию, выбрать грань, задать 2 параметра, выполнить). При выполнении операции над множеством граней необходимо выполнить $4 + X$ действий, где X - количество граней. Следовательно, уменьшение количества действий составит:

$$\text{абсолютное значение} = (1+p) * e - (p+e),$$

$$\text{относительное значение} = \frac{p+e}{(1+p)*e},$$

где p – количество заданных параметров; e – количество граней.

Конкретные значения для 2 заданных параметров представлены в таблице 3.3.

Таблица 3.3 - Уменьшение количества действий для рекомендации «Не наклоняйте каждую грань в отдельности»

Количество граней	Количество действий без использования рекомендации	Количество действий с использованием рекомендации	Абсолютное значение уменьшения количества действий	Относительное значение уменьшения количества действий, %
2	10	6	4	40
3	15	7	8	53
4	20	8	12	60
5	25	9	16	64

При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для четырёх и более объектов воспользуйтесь опцией «Запомнить состояние».

Область действия: операция «Равенство радиусов».

Условие: множество объектов (дуги и окружности) параметризованных попарно операцией «Равенство радиусов» не имеют центрально узла. То есть граф, в котором узлы - дуги и окружности, а отношения - операции «Равенство радиусов», не должен являться графом-звездой. Примеры графов представлены на рисунке 3.1, рисунке 3.2.

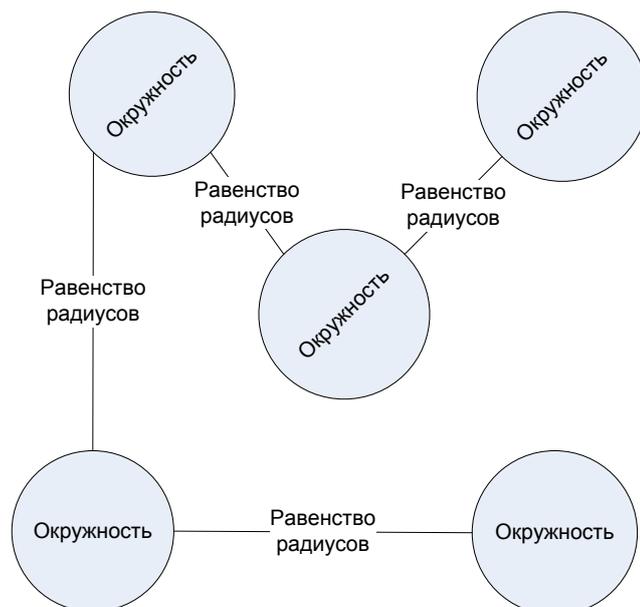


Рисунок 3.1 - Не рекомендованное построение параметрического ограничения «равенство радиусов двух дуг или окружностей»

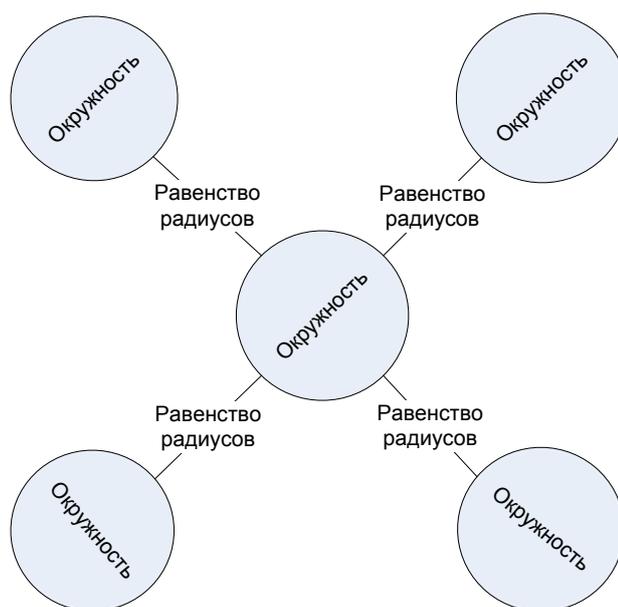


Рисунок 3.2 - Рекомендованное построение параметрического ограничения «равенство радиусов двух дуг или окружностей»

Причина: выполнение операции «Равенство радиусов» с опцией «Запомнить состояние» позволяет установить равенство радиусов сразу для нескольких дуг и/или окружностей, что уменьшает количество необходимых действий по сравнению с попарным выполнением операций над дугами/окружностями отдельно.

Результат: для выполнения операции «Равенство радиусов» над парами объектов необходимо 3 действия (выбрать операцию, выбрать 1 объект, выбрать 2 объект). С опцией «Запомнить состояние»: выбрать операцию, выбрать опцию «Запомнить состояние», выбрать объекты.

Следовательно, уменьшение количества действий составит:

$$\text{абсолютное значение} = 3 * (e - 1) - (3 + e),$$

$$\text{относительное значение} = \frac{3 + e}{3 * (e - 1)},$$

где e – количество объектов.

Конкретные значения представлены в таблице 3.4.

Таблица 3.4 - Уменьшение количества действий для рекомендации «Установить равенство радиусов сразу для нескольких дуг и/или окружностей»

Количество объектов	Количество действий без использования рекомендации	Количество действий с использованием рекомендации	Абсолютное значение уменьшения количества действий	Относительное значение уменьшения количества действий, %
4	9	6	3	33
5	12	7	5	42
6	15	8	7	47
7	18	9	9	50

При построении параметрического ограничения «равенство длин» для четырёх и более объектов воспользуйтесь опцией «Запомнить состояние».

Область действия: операция «Равенство длин».

Условие: множество отрезков параметризованных попарно операцией «Равенство длин» не имеют центрально узла. То есть граф, в котором узлы - отрезки, а отношения - операции «Равенство длин», не должен являться графом-звездой. Примеры графов представлены на рисунках рисунке 3.3, рисунке 3.4.

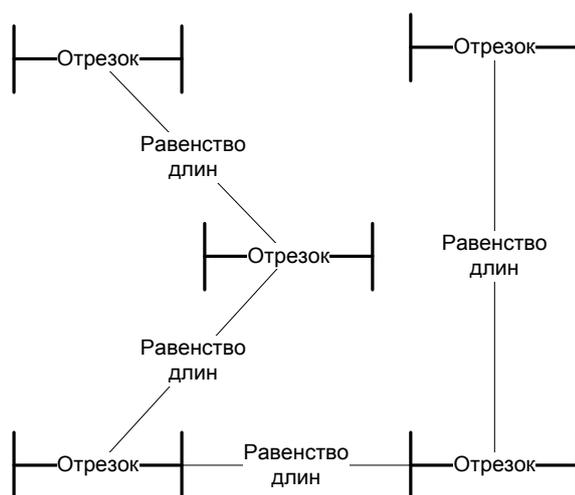


Рисунок 3.3 - Не рекомендованное построение параметрического ограничения «равенство длин»

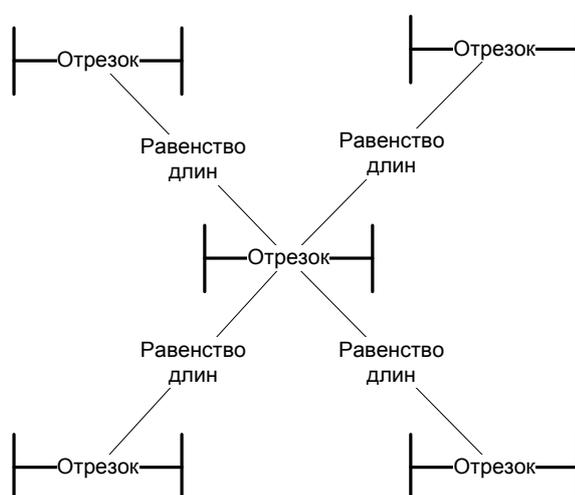


Рисунок 3.4 - Рекомендованное построение параметрического ограничения «равенство длин»

Причина: выполнение операции «Равенство длин» с опцией «Запомнить состояние» позволяет установить равенство радиусов сразу для нескольких отрезков, что уменьшает количество необходимых действий по сравнению с попарным выполнением операций над отрезками отдельно.

Результат: для выполнения операции «Равенство длин» над парами объектов необходимо 3 действия (выбрать операцию, выбрать 1 отрезок, выбрать 2 отрезок). С опцией «Запомнить состояние»: выбрать операцию, выбрать опцию «Запомнить состояние», выбрать отрезки.

Следовательно, уменьшение количества действий составит:

$$\text{абсолютное значение} = 3 * (e - 1) - (3 + e),$$

$$\text{относительное значение} = \frac{3+e}{3*(e-1)},$$

где e – количество объектов.

Конкретные значения представлены в таблице 3.5.

Таблица 3.5 - Уменьшение количества действий для рекомендации
«Установить равенство длин сразу для нескольких отрезков»

Количество отрезков	Количество действий без использования рекомендации	Количество действий с использованием рекомендации	Абсолютное значение уменьшения количества действий	Относительное значение уменьшения количества действий, %
4	9	6	3	33
5	12	7	5	42
6	15	8	7	47
7	18	9	9	50

Построенные тела располагаются в узлах параллелограммной сетки, воспользуйтесь операцией «Массив по сетке».

Область действия: формообразующие операции.

Условие: существуют тела с одинаковыми параметрами, расположение которых вписываются в узла параллелограммной сетки. Схема образования параллелограммной сетки представлена на рисунке 3.5.

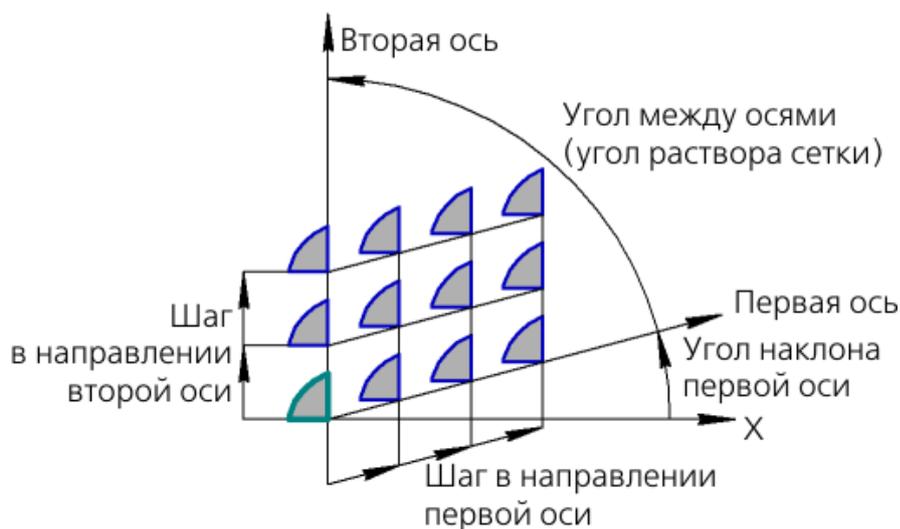


Рисунок 3.5 - Схема образования параллелограммной сетки [108]

Причина: выполнение операции «Массив по сетке» уменьшает количество объектов, что повышает производительность САПР, также уменьшает количество необходимых действий по сравнению с построением каждого тела отдельно.

Результат: для построения одного тела необходимо нарисовать эскиз и выполнить формообразующую операцию с заданными параметрами. При использовании операции «Массив по сетке» необходимо построить тело и дополнительно выполнить 15 действий (выбрать операцию, выбрать тело, задать 12 параметров, выполнить). Рассчитаем необходимое количество действий для каждого случая:

построение каждого тела отдельно: $bs * c$,

использование операции «Массив по сетке»: $bs + 2 + ps$,

где bs – количество действия для построения тела,

c – количество тел,

ps - количество заданных параметров для операции «Массив по сетке».

Конкретные значения для тела «Шайба» (для построения необходимо 6 действий) и для операции «Массив по сетке» с 4 параметрами представлены в таблице 3.6. Стоит отметить, что в «Шайба» - это простое тело (эскиз представлен окружностью), для других тел сокращение действий будет более значительным.

Таблица 3.6 - Уменьшение количества действий для рекомендации «Воспользуйтесь операцией Массив по сетке»

Количество тел	Количество действий без использования рекомендации	Количество действий с использованием рекомендации	Абсолютное значение уменьшения количества действий	Относительное значение, уменьшения количества действий %
3	18	13	5	28
4	24	13	11	46
5	30	13	17	57

Построенные тела располагаются в узлах концентрической сетки, воспользуйтесь операцией «Массив по концентрической сетке».

Область действия: формообразующие операции.

Условие: существуют тела с одинаковыми параметрами, расположение которых вписываются в узла concentрической сетки. Схема образования concentрической сетки представлена на рисунке 3.6.

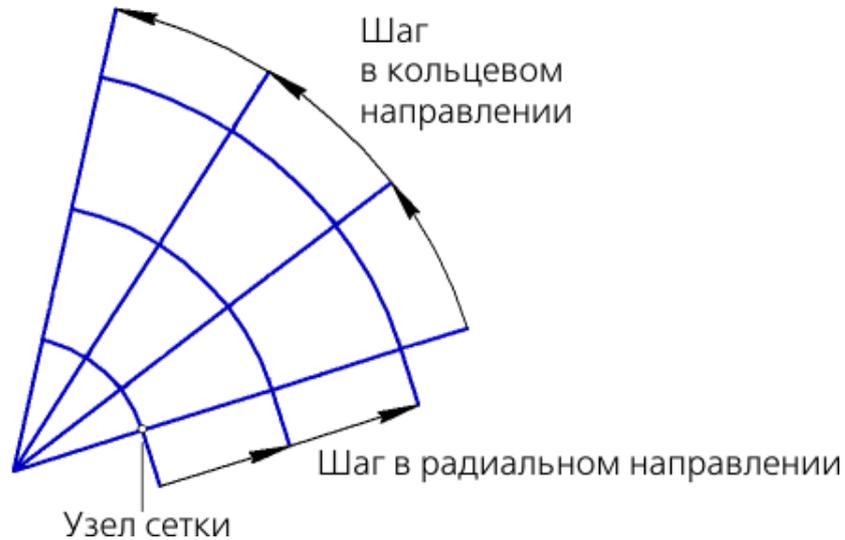


Рисунок 3.6 - Схема образования concentрической сетки [108]

Причина: выполнение операции «Массив по concentрической сетке» уменьшает количество объектов, что повышает производительность САПР, также уменьшает количество необходимых действий по сравнению с построением каждого тела отдельно.

Результат: для построения одного тела необходимо нарисовать эскиз и выполнить формообразующую операцию с заданными параметрами. При использовании операции «Массив по concentрической сетке» необходимо построить тело и дополнительно выполнить 11 действий (выбрать операцию, выбрать тело, выбрать ось, задать 7 параметров, выполнить). Рассчитаем необходимое количество действий для каждого случая:

построение каждого тела отдельно: $bs * c$,

использование операции «Массив по concentрической сетке»: $bs + 2 + ps$,

где bs – количество действия для построения тела; c – количество тел;

ps - количество заданных параметров для операции «Массив по concentрической сетке».

Конкретные значения для тела «Шайба» (для построения необходимо 6 действий) и для операции «Массив по концентрической сетке» с 4 параметрами представлены в таблице 3.7.

Таблица 3.7 - Уменьшение количества действий для рекомендации «Воспользуйтесь операцией Массив по концентрической сетке»

Количество тел	Количество действий без использования рекомендации	Количество действий с использованием рекомендации	Абсолютное значение уменьшения количества действий	Относительное значение уменьшения количества действий, %
3	18	14	4	22
4	24	14	10	42
5	30	14	16	53

3.3. Классификация рекомендаций

Выделим следующие способы уменьшения количества действий при построении проектного решения:

- исключение действий, которые не влияют на проектное решение:
 - возврат проектного решения в одно из предыдущих состояний:
 - удалённые операции,
 - возврат параметров операции в одно из предыдущих состояний;
 - действия, не изменяющие выполнение операций:
 - изменение параметров, не влияющих на результат операции;
 - повторные изменения параметров операции.
 - выполнение однотипных действий над группами объектов;
 - применение специальных операций для выполнения типовых задач (фаска, скругление и т.д.);
 - выстраивание оптимальной последовательности операций с целью уменьшения количества действий необходимых для переходов между операциями;
- Рассмотрим классификацию неоптимальных действий.

Удалённые действия

В проектном решении выполнены лишние операции, которые в дальнейшем были удалены.

Шаблон для поиска: операция удаления и удаленные операции. Результат: пустое множество.

Действия, кроме последних, по редактированию параметров операции

Некоторые операции в проектном решении редактировались по несколько раз. Задавайте сразу верные значения параметров операции, это уменьшит количество выполняемых действий.

Шаблон для поиска: операции с изменёнными параметрами. Результат: операции с последними значениями параметров.

Действия по редактированию параметров, не влияющих на результат операции

Значение параметра операции были изменены и отличаются от значения по умолчанию, хотя изменение параметра не влияет на результат операции.

Шаблон для поиска: операции со значениями параметров отличающихся от первоначальных и результаты операций не изменились. Результат: операции с первоначальными значениями параметров.

Повторяющиеся одиночные действия над объектами

В САПР КОМПАС отсутствуют.

Действия, которые можно заменить специальными операциями (фаска, скругление и т.д.)

Типовые операции: фаска, скругление, уклон, массивы и т.д.

Рассмотрим пример.

Построенные четыре отрезка имеют ограничения «Совпадение двух точек», «Горизонталь», «Вертикаль». Воспользуйтесь операцией «Прямоугольник».

Шаблон для поиска: 4 операции «отрезок» с попарными последовательными ограничениями «Совпадение двух точек», 1 и 3 отрезки имеют ограничение «Вертикаль» и равны между собой, 2 и 4 отрезки имеют ограничение «Горизонталь» и тоже равны между собой. Результат: операция «прямоугольник».

Другие примеры приведены в приложении 2.

Действия для переходов между операциями из-за неоптимальной последовательности операций

Рассмотрим пример.

При первом редактировании эскиза были построены не все геометрические объекты.

Шаблон для поиска: операции по построению геометрических объектов в определённом эскизе перемешаны с операциями вне данного эскиза. Результат: упорядоченные операции, сначала связанные с данным эскизом, потом остальные.

Другие примеры приведены в приложении 2.

3.4. Разработка метода формирования рекомендаций

Исходными данными для формирования рекомендаций является поток операций проектировщика. Закодируем этот поток как последовательность выполняемых проектных операций, модель которой имеет вид:

$$PrOper = (Operations, TypeOperation, ParamKey, ParamValue),$$

где $Operations = \{o_i | i = 1..k\}$ – множество операций,

$TypeOperation = \{tp_i | i = 1..TP\}$ – множество типов операций в САПР (например, «выдавливание», «вращение»),

$ParamKey = \{pk_i | i = 1..PK\}$ – множество ключей параметров операций,

$ParamValue = \{pv_i | i = 1..PV\}$ – множество значений параметров операций.

Модель операции имеет вид:

$$Operation = (id, type, number, pvo),$$

где id – уникальный идентификатор операции,

$type \in TypeOperation$ – тип операции,

$number$ – номер операции,

pvo – множество параметров операции со значением.

Модель параметра операции со значением имеет вид:

$$PVO = (key, value),$$

где $key \in ParamKey$ – название параметра,

$value \in ParamValue$ – значение параметра.

Модель исходных данных для формирования рекомендаций имеет вид:

$$S = (Operations, Rules, A, F_atom),$$

где $Operations = \{o \in Operation\}$ – множество проектных операций,

$Rules = \{r_i | i = 1..k\}$ – множество правил для поиска и замены не оптимальных проектных операций,

$A = \{a_i | i = 1..n\}$ – множество атомов знаний,

$F_atom = Operation \rightarrow A$ – функция отображения операции на атомы знаний.

Модель правила имеет вид:

$$Rule = (tpl, result),$$

где $tpl = \{t_i | i = 1..k\}$ – формула логики первого порядка [92] для поиска в протоколе проектных операций;

$result = \{res_i | i = 1..n\}$, $res = (C, key, value)$ – множество оптимальных проектных операций, заданных как упорядоченные тройки (код операции, параметр операции или множество параметров операции, значение параметра операции или множество значений), где код – константа, а параметр операции и значение параметра операции – формулы логики первого порядка.

Для формулы логики первого порядка зададим алфавит символов:

- предметные переменные:
 - $X = \{xp \in TypeOperation\}$ – множество операций,
 - $P = \{p \in ParamKey\}$ – множество ключей параметров операций,
 - $T = \{t \in ParamValue\}$ – множество значений параметров операций;
- символы логических операций: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$;
- кванторы: \forall, \exists ;
- вспомогательные символы: «()», «{ }» — скобки; «,» — запятая;
- термы:

- $type = Operations \rightarrow TypeOperation$ – определение типа операции,
- $code = Operations \rightarrow \mathbb{N}$ – определение кода операции,
- $number = Operations \rightarrow \mathbb{N}$ – определение номера операции,
- $param = Operations \times ParamKey \rightarrow ParamValue$ – определение значения параметра операции,
- $param_start = Operations \times ParamKey \rightarrow ParamValue$ – значение по умолчанию параметра операции,
- арифметические действия: $\mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ – умножение, сложение, вычитание, деление;
- предикаты:
 - 2-арные функции: больше ($>$), меньше ($<$), равно ($=$), не равно (\neq), \in , eq_op – равенство результатов операций.

Модель рекомендации имеет вид:

$$R = (op_before, op_after, F_steps, F_message),$$

где $op_before \subset Operations$ - множество не оптимальных проектных операций, задействованных при формировании рекомендации,

$op_after \subset Operations$ – множество рекомендуемых проектных операций,

$F_steps = Operations \rightarrow \mathbb{N}$ – функция вычисления количества действий для построения проектных операций,

$F_message = Operations \times Operations \rightarrow Text$ – функция формирования текста рекомендации.

Рассмотрим пример формирования рекомендации при построении детали «Кольцо». На рисунке 3.7 отображена последовательность операций с параметрами выполненных проектировщиком, и он готовится выполнить последнюю операцию «Фаска».

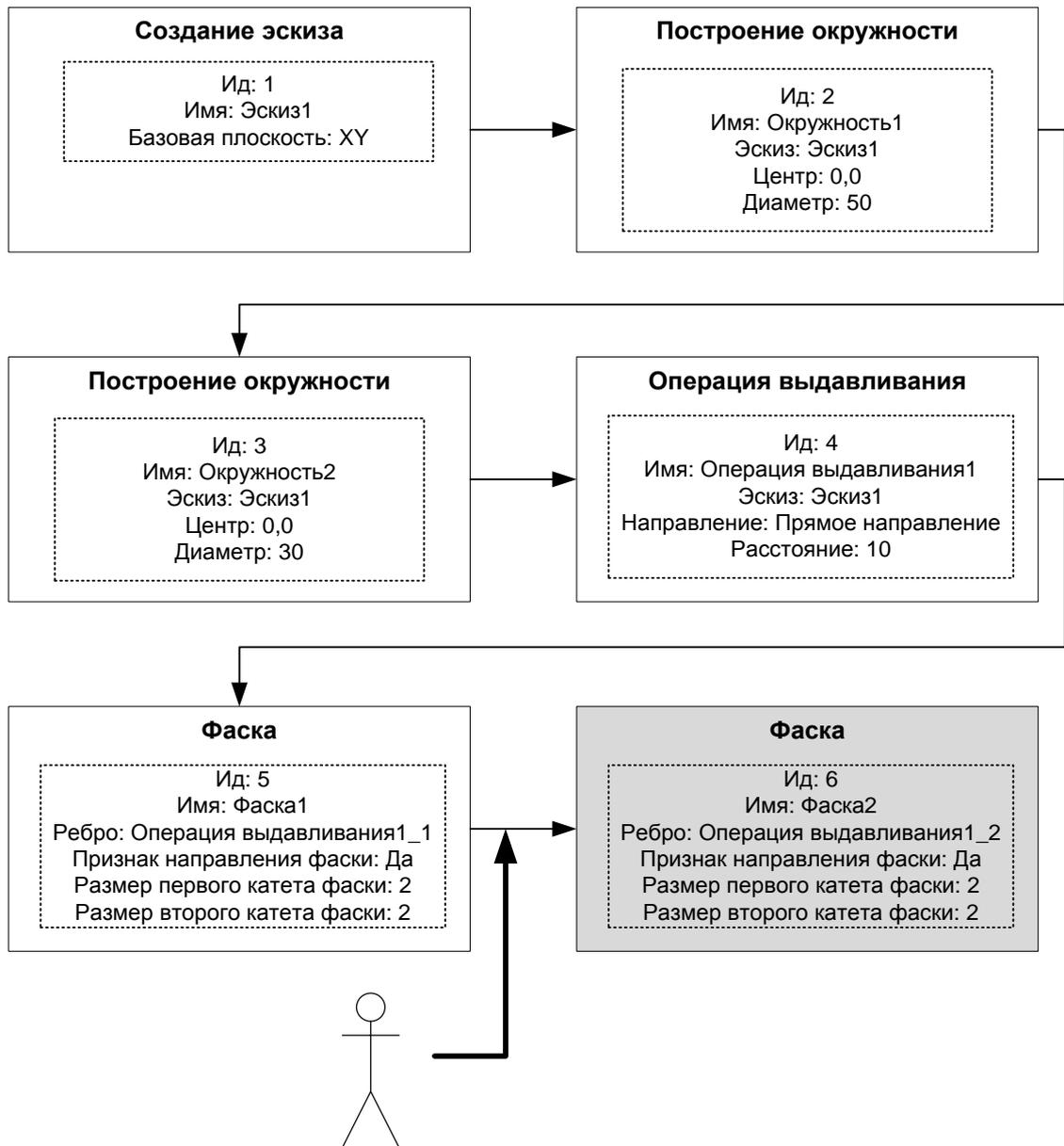


Рисунок 3.7 - Последовательность операций при построении детали «Кольцо»

После выполнения данной операции запускается поиск по базе правил, результатом которого является правило «Не стройте фаску для каждого ребра в отдельности». Шаблон этого правила имеет вид:

$$\begin{aligned}
 & type(xp_1) = \text{Фаска} \wedge type(xp_2) = \text{Фаска} \wedge \\
 & param(xp_1, \text{Признак направления}) = param(xp_1, \text{Признак направления}) \wedge \\
 & param(xp_2, \text{Размер первого катета}) = param(xp_2, \text{Размер первого катета}) \wedge \\
 & param(xp_2, \text{Размер второго катета}) = param(xp_2, \text{Размер второго катета}).
 \end{aligned}$$

А формула для построения множества оптимальных проектных операций имеет вид:

$$\left\{ \begin{array}{l} (1, \text{Тип, Фаска}), \\ (1, \text{Признак направления}, \text{param}(xp_1, \text{Признак направления})), \\ (1, \text{Размер первого катета}, \text{param}(xp_1, \text{Размер первого катета})), \\ (1, \text{Размер второго катета}, \text{param}(xp_1, \text{Размер второго катета})), \\ (1, \text{Ребро}, \text{param}(xp_1, \text{Ребро})), (1, \text{Ребро}, \text{param}(xp_2, \text{Ребро})) \end{array} \right\}$$

Графическое представление данного правила показано на рисунке 3.8.

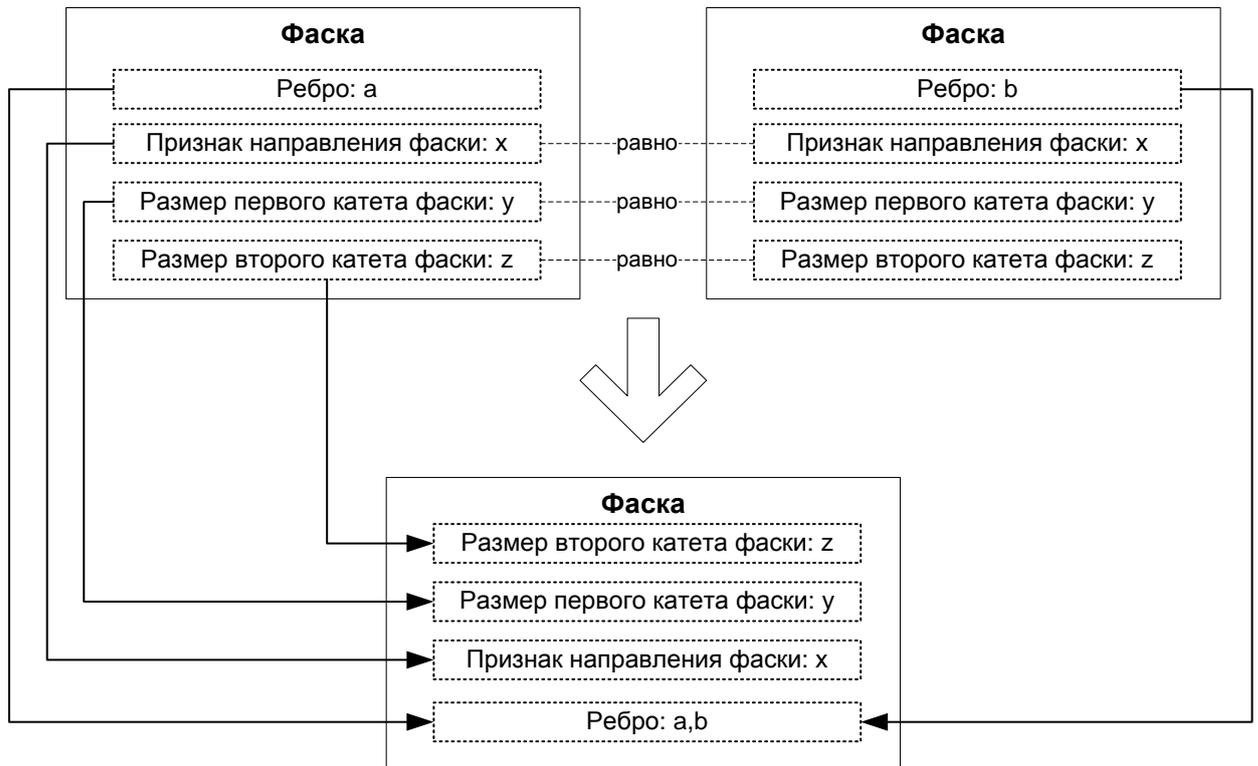


Рисунок 3.8 - Графическое представление для правила «Не стройте фаску для каждого ребра в отдельности»

Результат применения правила для нашей последовательности операций показан на рисунке 3.9.

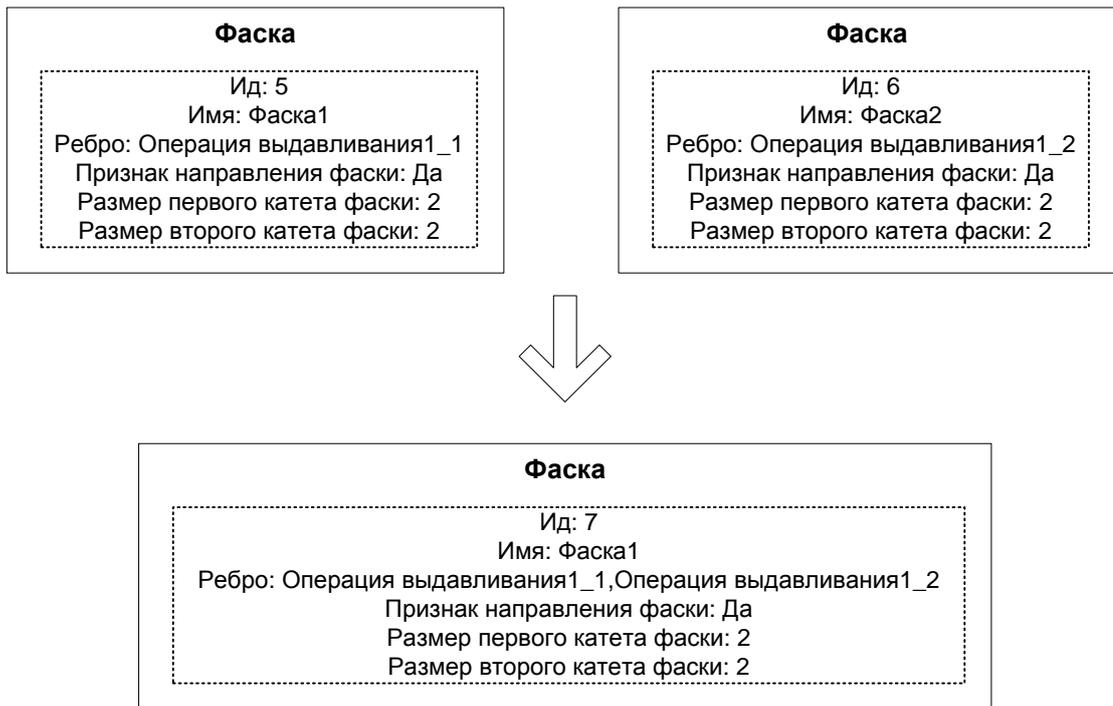


Рисунок 3.9 - Результат применения правила «Не стройте фаску для каждого ребра в отдельности»

Для данного случая количество выполняемых действий проектировщиком сократилось с 12 до 7. Профиль проектировщика скорректируется следующим образом:

$$p_{ab} = \frac{p_{ab} * c_{ab} - 1}{c_{ab} + 1},$$

$$c_{ab} = c_{ab} + 1,$$

где c_{ab} – степень владения умением «построение фаски», p_{ab} - количество контрольных измерений данного умения. Также изменится степень владения навыком и среднее время выполнения операции.

Алгоритм формирования рекомендаций и корректировки профиля проектировщика представлен ниже.

1. Начало работы проектировщика с проектом.
2. Если проект новый переход к шагу 5.
3. Генерация операций на основе имеющегося проекта.
4. Добавление операции в последовательность операций.
5. Считывание управляющего воздействия от проектировщика.
6. Генерация операций на основе действий проектировщика.

7. Добавление операции в последовательность операций.
8. Поиск правила, которое соответствует последовательности операций.
9. Если правило не найдено переход к шагу 11.
10. Формирование оптимального множества операций.
11. Формирование рекомендации на основе множеств оптимальных и неоптимальных операций.
12. Добавление рекомендации в индивидуальный список рекомендаций проектировщика и вывод на экран.
13. Корректировка умений и навыков проектировщика соответствующие задействованным операциям.
14. Если работа с проектом не закончена переход к шагу 5.
15. Выход.

3.5. Разработка модели виртуального компонента автоматизированной обучающей системы

Предлагаемая модель виртуального тренажёра разработана на базе автоматного подхода. Отличительной особенностью модели является наличие функциональных блоков и ошибочных состояний, позволяющих расширить класс диагностируемых ошибок обучаемого.

Модифицированная модель виртуального тренажёра на базе автоматного подхода.

Для решения проблемы диагностики ошибок обучаемого при работе со сложными объектами разработана модель виртуального тренажёра [62]:

Тренажёр = (множество состояний, множество управляющих воздействий, множество правил, множество функциональных блоков, множество ошибочных состояний),

где *множество состояний* - состояния, полученные в пространстве параметров тренажёра;

множество управляющих воздействий - набор возможных пользовательских действий с тренажёром;

множество правил - условия перехода тренажёра из одного состояния в другое;

множество функциональных блоков - функциональный блок - это обособленная часть тренажёра, которая выполняет свою самостоятельную задачу и не пересекается с остальными блоками;

множество ошибочных состояний - множество состояний, которые характеризуют неработоспособность тренажёра.

Правило = (*состояние, управляющее воздействие, следующее состояние*), где *состояние* - состояние, в котором должен находиться тренажёр для применения правила;

управляющее воздействие - действие обучаемого, которое инициирует смену состояния;

следующее состояние - состояние, в которое перейдёт тренажёр, после применения правила.

Функциональный блок = (*множество правил, множество целевых правил*), где *множество правил* - правила, которые выполняются в рамках работы с данным функциональным блоком;

множество целевых правил - у каждого блока есть своя цель (например, установить режим запуска), последнее правило, которое применяется для достижения цели блока, и есть целевое правило.

На рисунке 3.10 представлен пример работы виртуального тренажёра генератора импульсов точной амплитуды Г5-75.

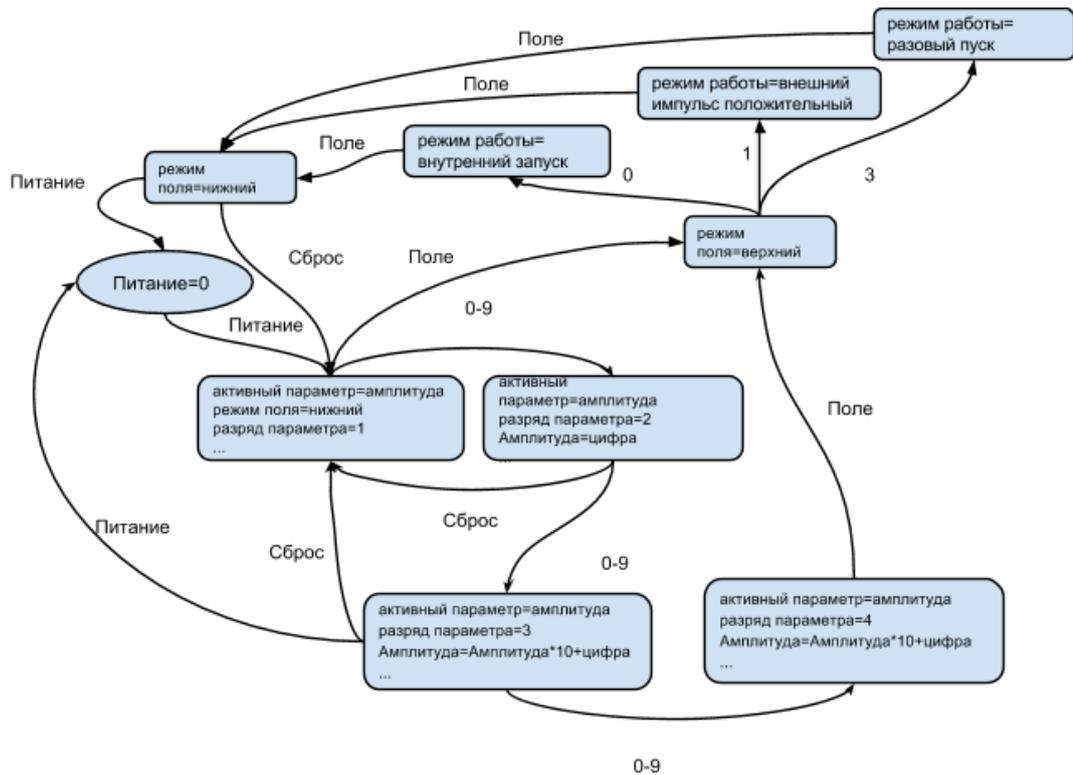


Рисунок 3.10 - Пример работы виртуального тренажёра генератора импульсов точной амплитуды Г5-75[64]

Алгоритм построения тренажёра



Рисунок 3.11 - Алгоритм построения тренажёра [68]

Представлено содержательное описание алгоритма построения тренажёра (рисунок 3.11).

Описываемая модель успешно апробирована на практике, среди разработанных тренажёров: осциллограф С1-116, осциллограф С1-127, осциллограф С1-125, таймер/счетчик/анализатор CNT-90, генератор сигналов высокочастотный РГ4-17-01А, генератор сигналов высокочастотный Г4-82, блок преобразования частоты автоматический ЯЗЧ-175 (ЯЗЧ-175/1), частотомер электронно-счетный вычислительный ЧЗ-64, генератор импульсов точной амплитуды Г5-75[16], лабораторные стенды "Измерение матрицы рассеяния СВЧ устройства", "Поляризатор СВЧ", "Согласование линии передачи с устройством СВЧ". Более подробно с разработками можно ознакомиться в работах [100, 61].

На рисунке 3.12 показана архитектура тренажёра.

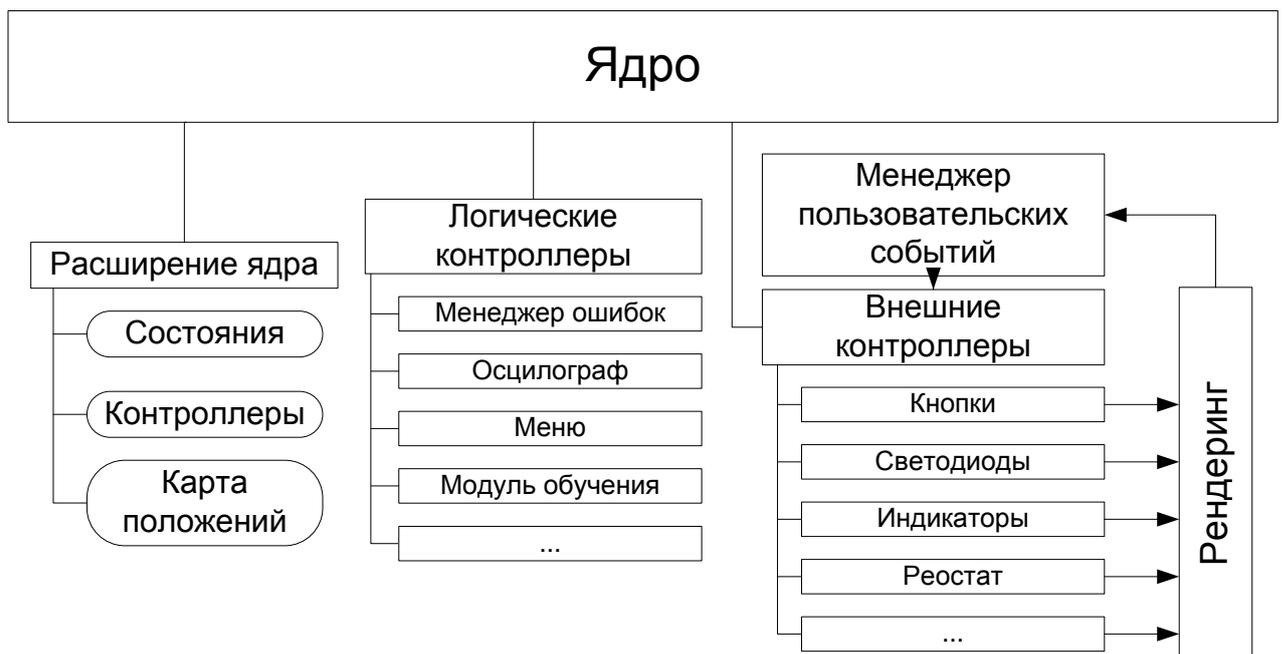


Рисунок 3.12 - Архитектура тренажёра [65]

3.6. Разработка метода автоматической диагностики ошибок обучаемого при работе с виртуальными компонентами автоматизированной обучающей системы

Предлагаемый метод анализа действий и диагностики ошибок обучаемого работает полностью в автоматическом режиме без помощи эксперта, а также

поддерживает свободу выбора решения у обучаемого, определяет тип ошибки из 7 возможных, обеспечивает нейтрализацию ошибки, то есть продолжение поиска других ошибок.

Для проверки навыков работы с тренажёром обучаемому выдаются упражнения. Они характеризуются начальным и конечными состояниями:

упражнение = (Начальное состояние, конечные состояния),

где *начальное состояние* - состояние, с которого начинается выполнение упражнения;

конечные состояния - состояния, в которых упражнение считается выполненным.

В процессе выполнения упражнения обучаемый генерирует управляющее воздействие на тренажёр. Запустим счётчик тактов, каждое управляющее воздействие увеличивает счётчик на 1. Управляющее воздействие инициирует смену состояния тренажёра. Состояние тренажёра после такта t обозначим как $S(t)$. Применяемое правило для перехода из состояния $S(t-1)$ в $S(t)$ обозначим как $R(t)$.

Проведём классификацию ошибок обучаемых и разработаем правила для их определения [63].

Петля:

$$S(t) == S(t-n), n=1...t.$$

Возвращение к состоянию, которое было в прошлом, частный случай - отсутствие изменения состояния после управляющего воздействия.

Неизменное состояние:

$$S(t) == S(t-1).$$

Управляющее воздействие не изменяет состояние тренажёра. Состояние тренажёра равно предыдущему, частный случай петли.

Ошибка тренажёра:

$$S(t) \in \text{множеству ошибочных состояний}.$$

Пользователь перевёл тренажёр в неисправное состояние. Текущее состояние принадлежит множеству ошибочных состояний.

Повторное достижение цели:

$\exists b \in \text{множеству функциональных блоков } \wedge R(t) \in \text{множеству целевых правил блока } b \wedge \exists n \neq t: R(n) \in \text{множеству целевых правил блока } b.$

Пользователь второй раз поменял какой-либо параметр, например, выставил сначала период на 10 секунд, затем на 8 секунд.

Переход между блоками:

$\exists b1, b2 \in \text{множеству функциональных блоков } \wedge R(t) \in \text{множеству правил блока } b1 \wedge \exists n \neq t: R(n) \in \text{множеству целевых правил блока } b2.$

Пользователь сначала выполнял задачу в рамках одного функционального блока, затем на полпути перешёл на выполнение цели в рамках другого блока. Например, начал устанавливать период в 15 секунд, не закончил и перешёл на включение режима внутреннего запуска.

Повторное использование функционального блока:

$\exists b \in \text{множеству функциональных блоков } \wedge R(t) \in \text{множеству правил блока } b \wedge \exists n \neq t: R(n) \in \text{множеству целевых правил блока } b.$

Пользователь начал менять какой-либо параметр, который уже был выставлен до этого.

Использование функционального блока без достижения цели:

$\exists b \in \text{множеству функциональных блоков}, \exists k: R(k) \in \text{множеству правил блока } b \wedge \nexists n: R(n) \in \text{множеству целевых правил блока } b \wedge S(t) \in \text{множеству конечных состояний данного упражнения}.$

Пользователь выполнил лишнее действие, которое никак не влияет на результат упражнения. Например, начал настраивать амплитуду сигнала, когда для успешного завершения упражнения этого не требуется.

На рисунке 3.13 представлено содержательное описание алгоритма поиска ошибочных действий обучаемого [2].

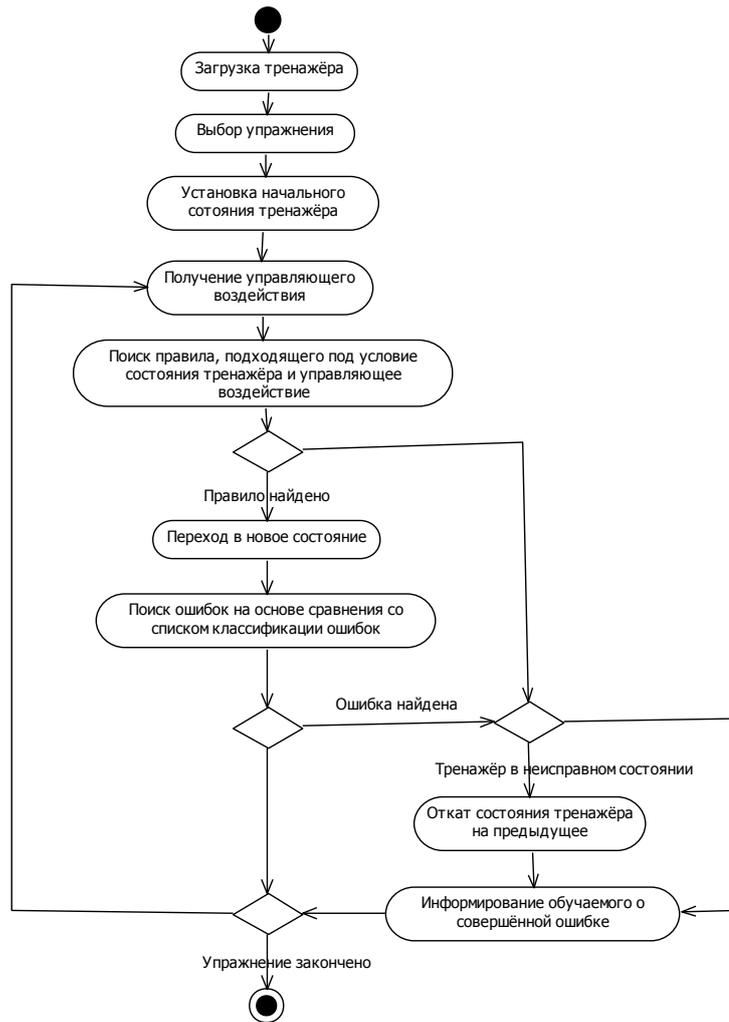


Рисунок 3.13 - Алгоритм поиска ошибочных действий обучаемого [65]

3.7. Выводы и рекомендации

1. Выделены следующие способы уменьшения количества действий при построении проектного решения: исключение действий, которые не влияют на проектное решение; выполнение однотипных действий над группами объектов; применение специальных операций для выполнения типовых задач (фаска, скругление и т.д.); выстраивание оптимальной последовательности операций с целью уменьшения количества действий, необходимых для переходов между операциями.

2. Разработан метод формирования рекомендаций и корректировки профиля компетенций проектировщика на основе протокола проектных операций, отличающийся анализом операций трехмерного моделирования деталей,

выполняемых в САПР КОМПАС, и позволяющий сформировать рекомендации и скорректировать профиль компетенций проектировщика.

3. Повышение эффективности деятельности проектировщика достигается за счёт поиска не оптимально выполненных проектных операций и рекомендацией замены их на операции с меньшим количеством действий. Уменьшение количества действий составляет от 17 до 40%. Исходными данными для формирования рекомендаций является поток действия проектировщика в САПР.

4. Правило для обработки протокола проектных операций состоит из двух частей: шаблон поиска неоптимального множества операций и генерация оптимального множества операций, описанные на основе логики первого порядка.

5. Рекомендации позволяют корректировать профиль проектировщика за счёт связи между выполненными операциями, навыками и умениями проектировщика.

6. Разработана модель виртуального тренажёра, отличающаяся наличием функциональных блоков и множеством ошибочных состояний, позволяющая формализовать процесс разработки тренажёра, накапливать библиотеку проектных решений и повторно использовать разработанные компоненты тренажёра.

7. С учётом предложенной модели разработан метод анализа действий обучаемого проектировщика, отличающийся использованием авторской модели тренажёра на базе автоматного подхода и шаблонов неэффективных действий, позволяющий автоматически оценить действия обучаемого.

8. Рекомендуется применение метода формирования рекомендаций и корректировки профиля компетенций проектировщика для оценки качества проектных решений.

Глава 4. Реализация компьютерной системы формирования профиля компетенций проектировщика

В данной главе перейдем к вопросу о реализации экспериментально-практической части диссертационной работы, а именно, к построению системы формирования профиля компетенций проектировщика.

Опишем и представим в виде UML (Unified Modeling Language) диаграмм общее представление функционального назначения системы и поведения, а также наиболее важных компонентов системы.

4.1. Разработка компонентной архитектуры системы формирования профиля компетенций проектировщика

СФПК построена по классической трёхзвенной архитектуре: клиент, сервер приложений и сервер баз данных.

В качестве платформы разработки используется технология Java Platform, Standard Edition [82], рассчитанная на индивидуальное пользование или на использование в масштабах малого предприятия. Имеются встроенные средства поддержки клиент-серверных приложений, например, Remote Method Invocation (RMI), Simple Object Access Protocol (SOAP).

В качестве сервера баз данных используется MySQL, вследствие бесплатности и популярности этой системы управления базами данных.

Реализуемая система является расширяемой, то есть поддерживает модельный принцип и позволяет расширять свою функциональность за счёт плагинов.

Рассмотрим более подробно две задачи построения системы: это реализация клиент-серверной технологии и поддержка плагинов.

Реализация клиент-серверной технологии

Для реализации клиент-серверной технологии [93] на Java Platform Standard Edition существует два альтернативных решения: RMI и веб-сервисы (SOAP) [78, 99, 54, 56]. Сравнительная таблица этих двух технологий представлена в таблице 4.1.

Таблица 4.1 - Сравнение RMI и SOAP

	RMI	SOAP
Скорость работы	Высокая	Низкая
Платформа	Java	Более 20 языков
Формат данных	Бинарная сериализация	XML
Соединение	Постоянное	Временное
Запрос клиенту	Да	Нет
Наличие коннектора	Обязательно	Нет

Выбор реализации клиент-серверной технологии пал в сторону веб-сервисов, основным преимуществом по сравнению с похожей технологией RMI, является независимость от платформы Java [75]. Архитектура этого решения представлена на рисунке 4.1.

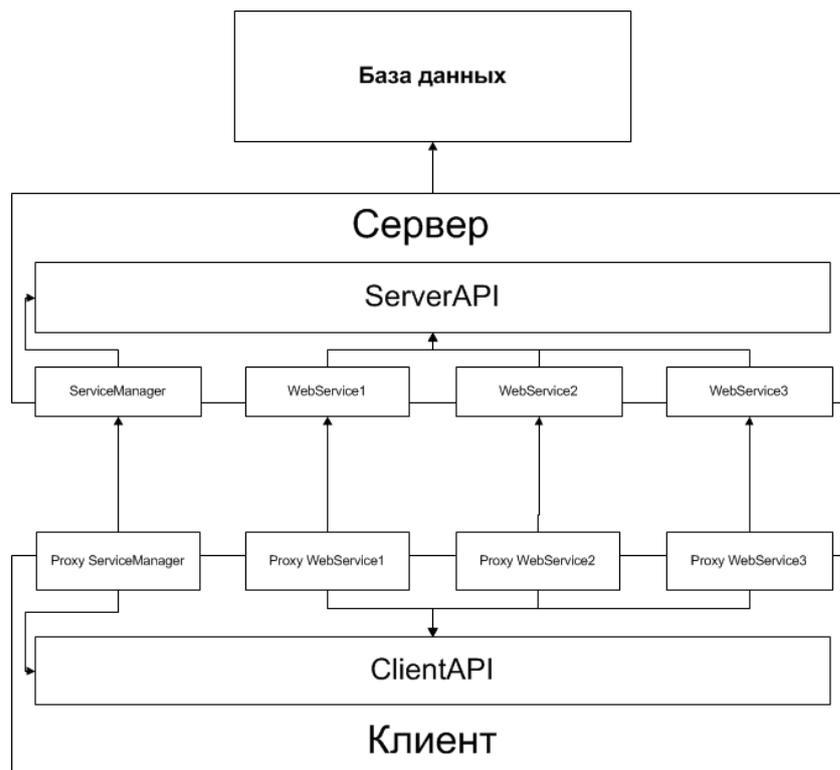


Рисунок 4.1 - Клиент-серверная расширяемая архитектура СФПК [17]

Для реализации веб-сервиса в Java необходимо выполнить следующие шаги.

1. Объявить класс с аннотацией `@WebService`, который указывает, что данный класс будет использоваться как веб-сервис.

2. Опубликовать класса, сгенерировать WSDL[10, 121] (Web Services Description Language). Таким образом происходит регистрация веб-сервиса в сети, а WSDL содержит описание службы, поддерживаемые методы и параметры.

3. Генерации заглушки по WSDL на клиенте, по WSDL-файлу Java создаёт локальный класс-заглушку, с которым работает клиент как с обычным java-классом.

4. Работа с заглушкой.

Реализация плагинов

Для расширения функциональности системы формирования профиля компетенций используется технология плагинов, которая позволяет динамически подключать дополнительные модули системы, тем самым расширяя её возможности. Плагины позволяют расширять как клиентскую, так и серверную часть, благодаря предоставлению клиентского и серверного API соответственно.

Плагины для своей работы используют технологию веб-сервисов. Каждый плагин должен реализовать клиентскую и серверную часть. Серверная часть реализует дополнительные веб-сервисы, с которыми работают клиенты. Клиентская часть расширяют функциональность системы за счёт работы с веб-сервисами, предоставленными сервером.

Реализованы следующие плагины системы (рисунок 4.2):

- редактор тестов;
- редактор сценариев [76];
- редактор моделей предметной области [77];
- модель обучаемого инженера;
- компонент «файл-менеджер» для трансфера [79];
- виртуальный учитель - веб-ориентированная система представления учебно-методического материала;
- клиентский модуль для управления пользователями.

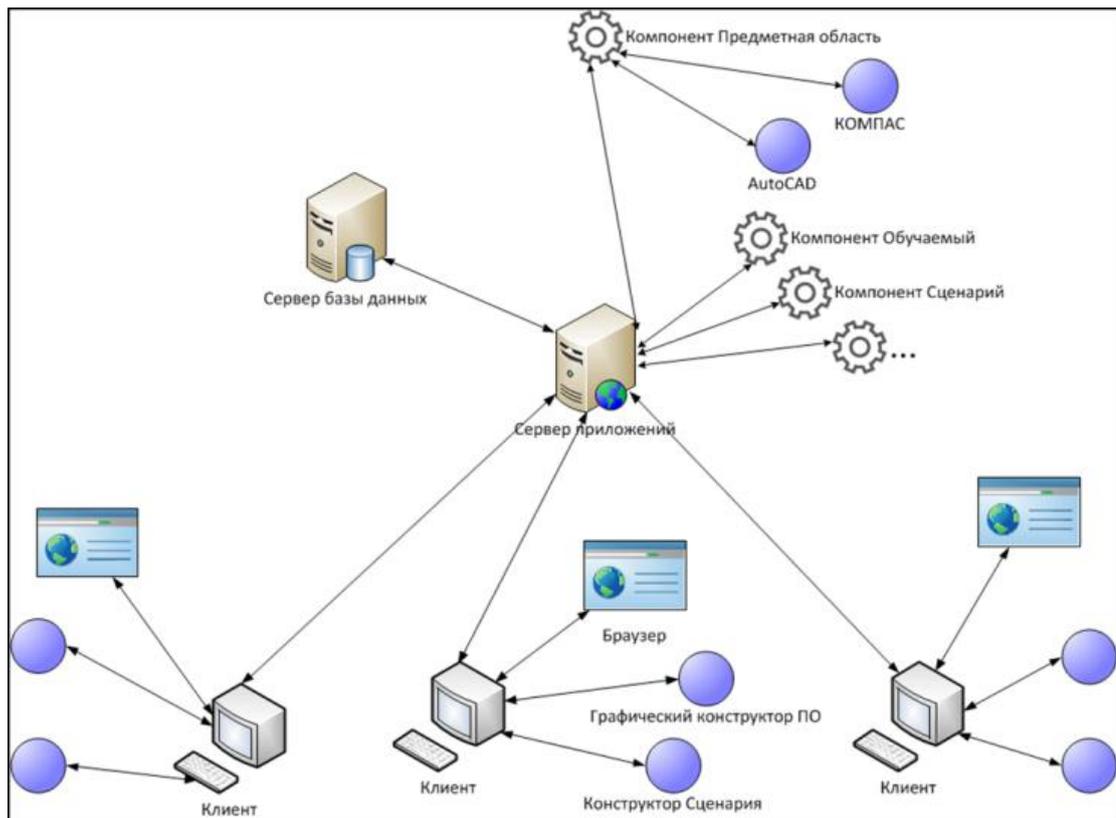


Рисунок 4.2 - Компонентная архитектура СФПК [78]

Рассмотрим некоторые особенности работы с подключением плагинов в Java. Для начала рассмотрим работу с плагинами в C#.

Для загрузки плагина используется метод `Assembly LoadFrom (string assemblyFile)`.

Загружает сборку .Net в текущий домен. Сборка загружается независимо от того, имеются ли неопределённые зависимости, проверка происходит непосредственно при работе с классом, например, при создании экземпляра класса. При этом зависимые классы ищутся в текущем домене автоматически, то есть среди других загруженных сборок. Таким образом, при наличии зависимостей между плагинами обычно не возникает никаких проблем, так как сначала загружаются все плагины, а только затем между ними строятся зависимости.

Если всё-таки сборка не найдена, вызывается событие `AppDomain.CurrentDomain.AssemblyResolve` в обработчике которого описывается, каким-образом необходимо загрузить ненайденную сборку.

Работа с плагинами в Java.

Сначала необходимо создать собственный `ClassLoader` - класс, унаследованный от `ClassLoader` с переопределёнными методами:

```
Class<?> loadClass (String name),
```

```
URL getResource (String name).
```

`Class<?> loadClass (String name)` вызывается, когда необходимо получить класс, в качестве параметра указывается имя класса.

`URL getResource (String name)` – получение ресурса класса по имени.

Для создания класса используется следующий метод

```
Class<?> defineClass (String name, byte [] b, int off, int len)
```

Таким образом, в Java в отличие от C# необходимо реализовать методы для загрузки сборок, самостоятельно решить проблему с зависимостями сборок, их нахождение и загрузку и переопределить метод для нахождения и загрузки ресурсов (картинки, файлы и т. д.)

4.2. Реализация графического редактора модели предметной области

Редактор реализован на основе программы «Графический конструктор модели предметной области САПР» [77, 67].

Внешний вид редактора предметной области представлен на рисунке 4.3. Программа разработана на платформе Java Platform Standart Edition и работает в связке с СУБД MySQL 5.1 [43].

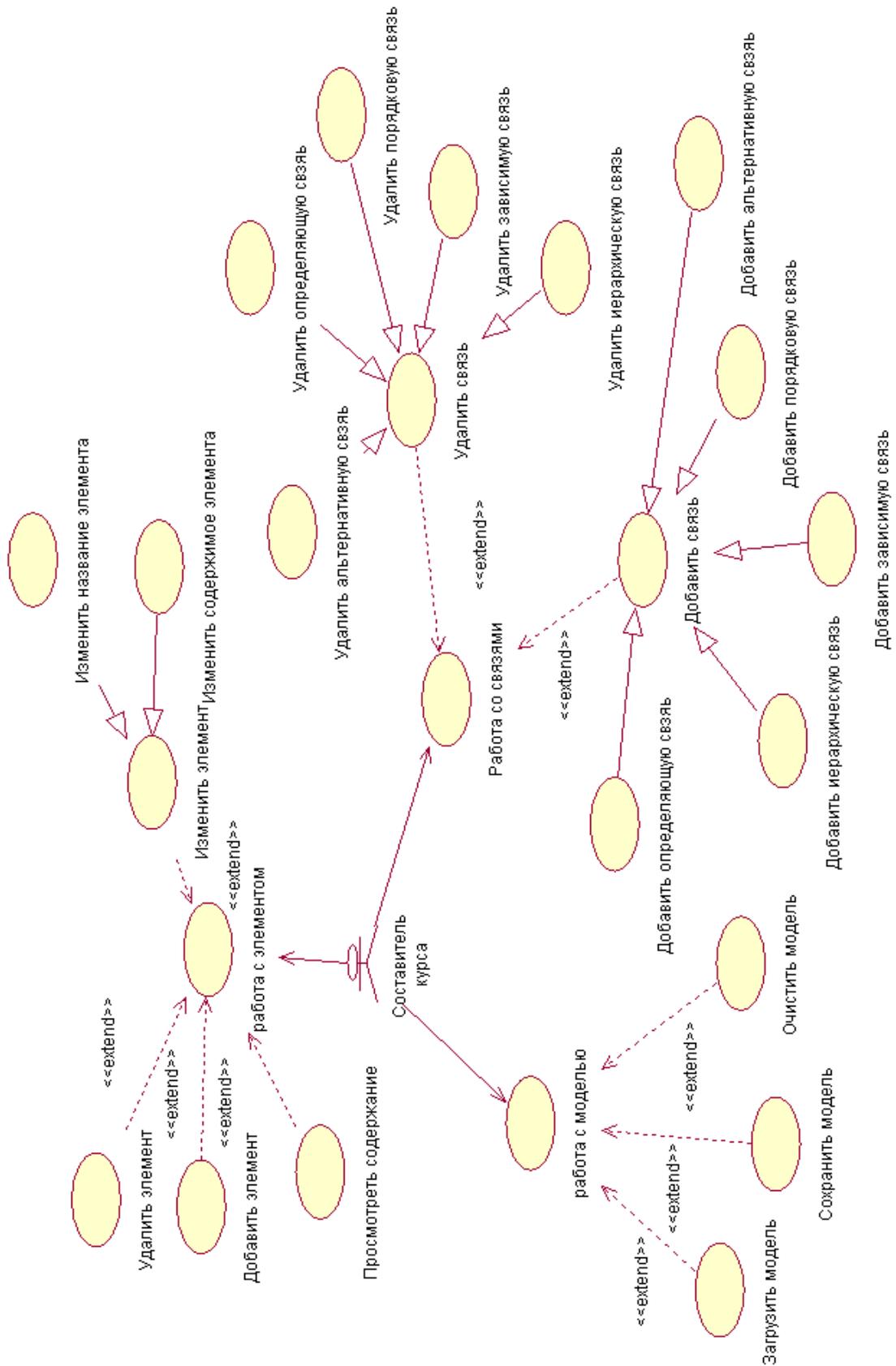


Рисунок 4.4 - Диаграмма использования для графического редактора модели предметной области [98]

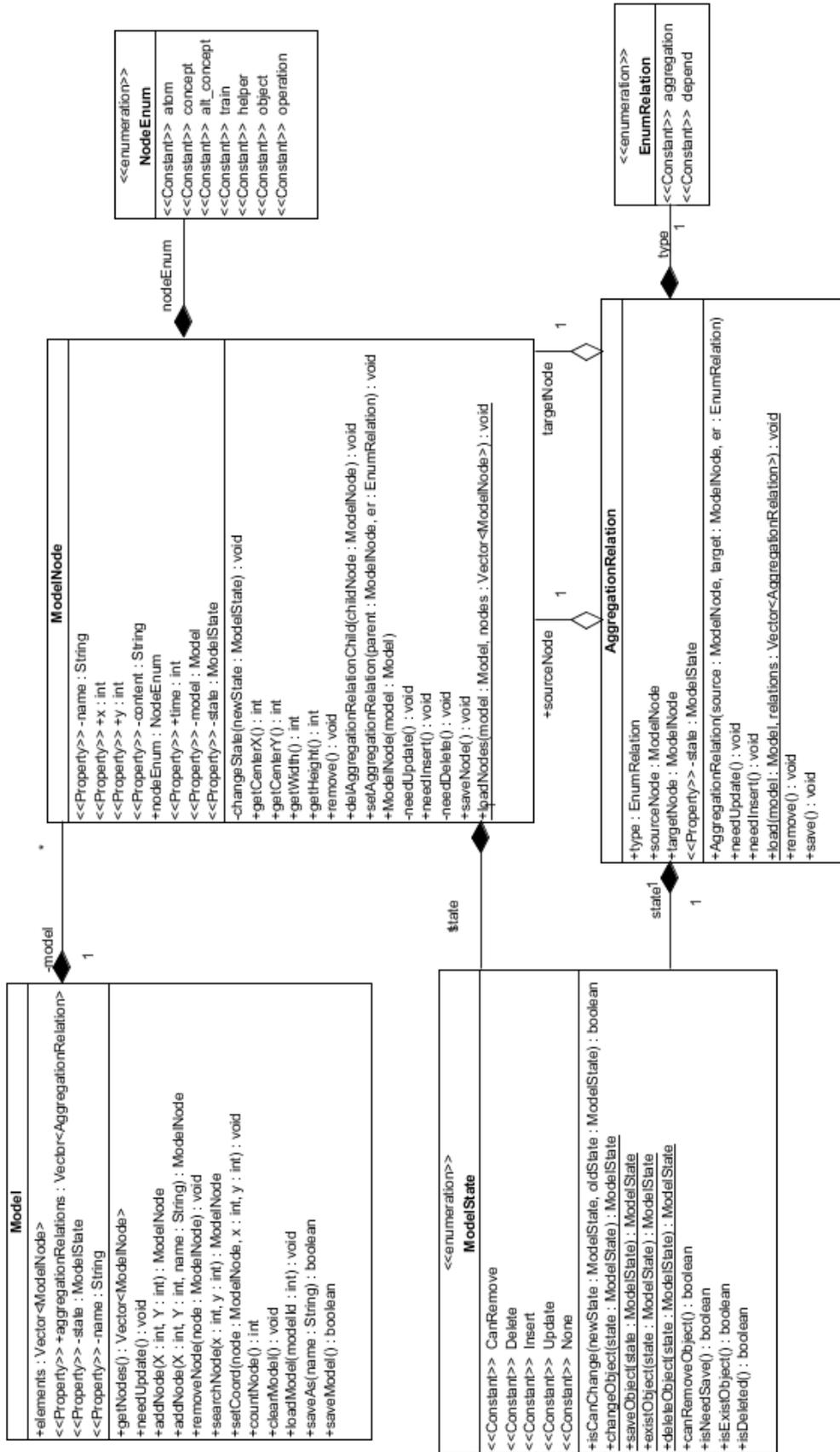


Рисунок 4.6 - Диаграмма классов для графического редактора модели предметной области [98]

4.3. Реализация метода формирования персонифицированного сценария обучения

Метод формирования персонифицированного сценария обучения реализован на языке программирования Java и включает следующие компоненты:

- виртуальный учитель;
- модель предметной области;
- модель сценария;
- модель тестирования;
- модель обучаемого инженера.

На рисунке 4.7 показан результат работы алгоритма, в том числе тестовое задание с вариантами ответов, выбранные ответы, степени владения знаниями у обучаемого проектировщика и сформированная последовательность материалов для изучения.

```

Окружность по двум точкам: 50%(0) Окружность по трем точкам: 50%(0)
Вопрос: Что такое окружность?
1) Определение окружности+
2) Отрезок, касательный к двум кривым
3) Касательный отрезок через точку кривой
4) Касательный отрезок из внешней точки
5) Окружность по двум точкам
6) Произвольный отрезок
Ответ: 2 1
Состояние атомов:
Касательный отрезок из внешней точки: 100%(1) Отрезок, касательный к двум кривым: 33%(3)
Касательный отрезок через точку кривой: 50%(2) Параллельный отрезок: 100%(1)
Перпендикулярный отрезок: 0%(1) Произвольный отрезок: 0%(1)
Определение отрезка: 100%(1) Определение окружности: 100%(1)
Окружность, касательная к двум кривым: 0%(1) Окружность, касательная к кривой: 0%(1)
Окружность по двум точкам: 50%(0) Окружность по трем точкам: 50%(0)
Материал для изучения:
Построение окружности, Построение отрезков,

```

Рисунок 4.7 - Результат работы алгоритма [98]

Диаграмма использования (рисунок 4.8) содержит двух актёров: обучаемого и виртуального учителя, а также методы их взаимодействия. Представлена диаграмма деятельности (рисунок 4.9) и классов (рисунок 4.10).

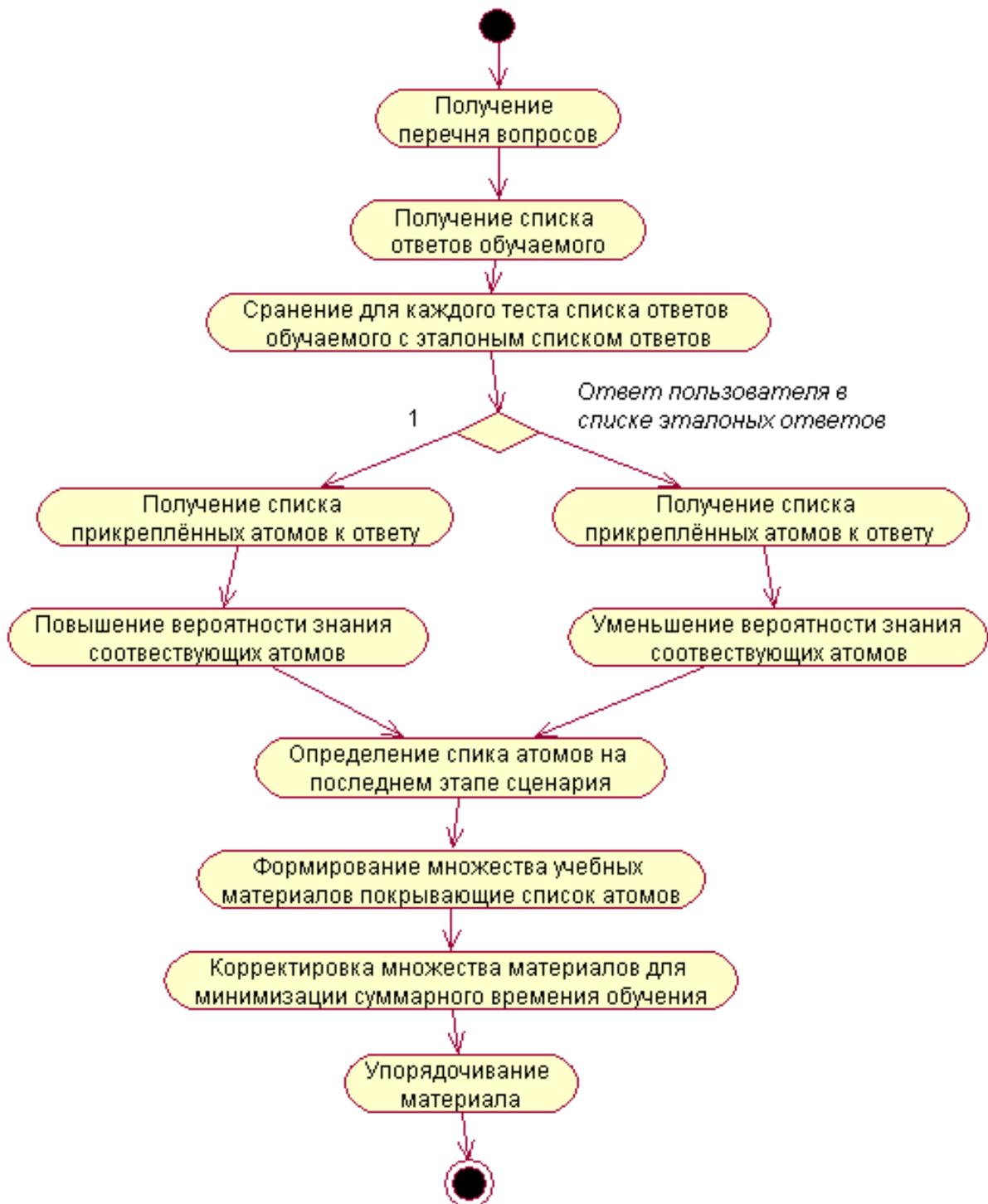


Рисунок 4.9 - Диаграмма деятельности [98]

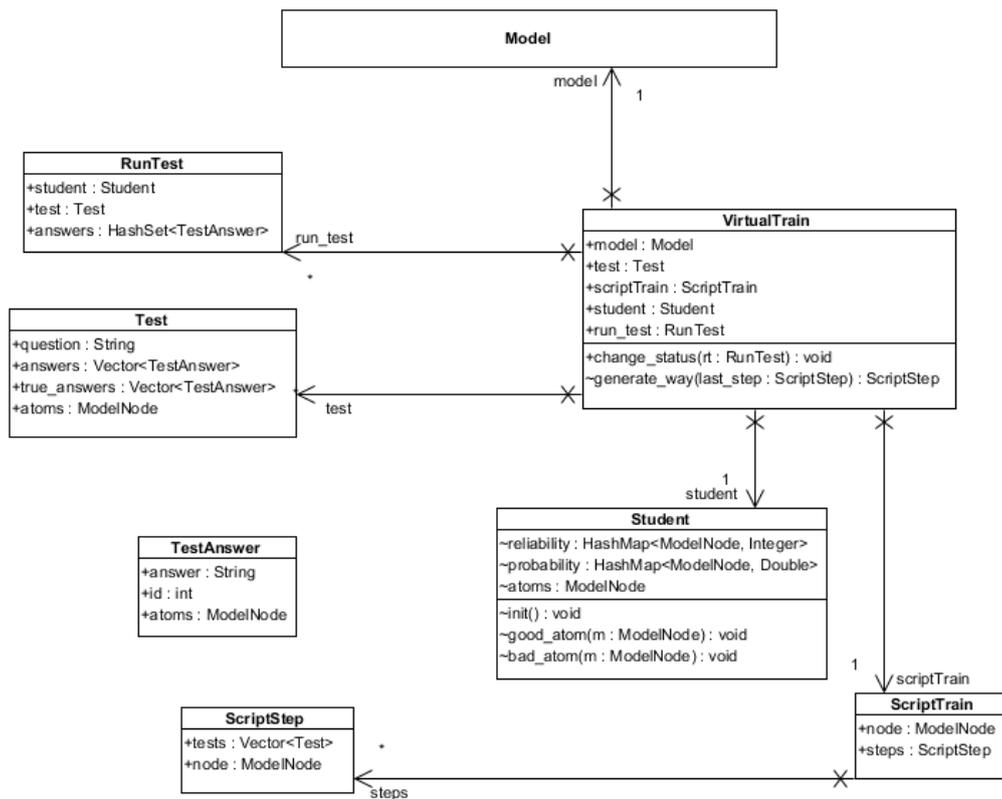


Рисунок 4.10 - Диаграмма классов [98]

4.4. Реализация системы рекомендаций для проектировщика

Система рекомендаций состоит из 2 основных частей: генератор операций и экспертная система по поиску рекомендаций (рисунок 4.11).

Генератор операций

Генератор операций реализован на платформе .NET Framework [112]. В качестве примера выбрана система автоматизированного проектирования КОМПАС-3D V16. Для обработки событий используется технология Automation [109], реализованная в виде библиотеки на языке программирования C#. Компоненты генератора приведены в таблице 4.2.

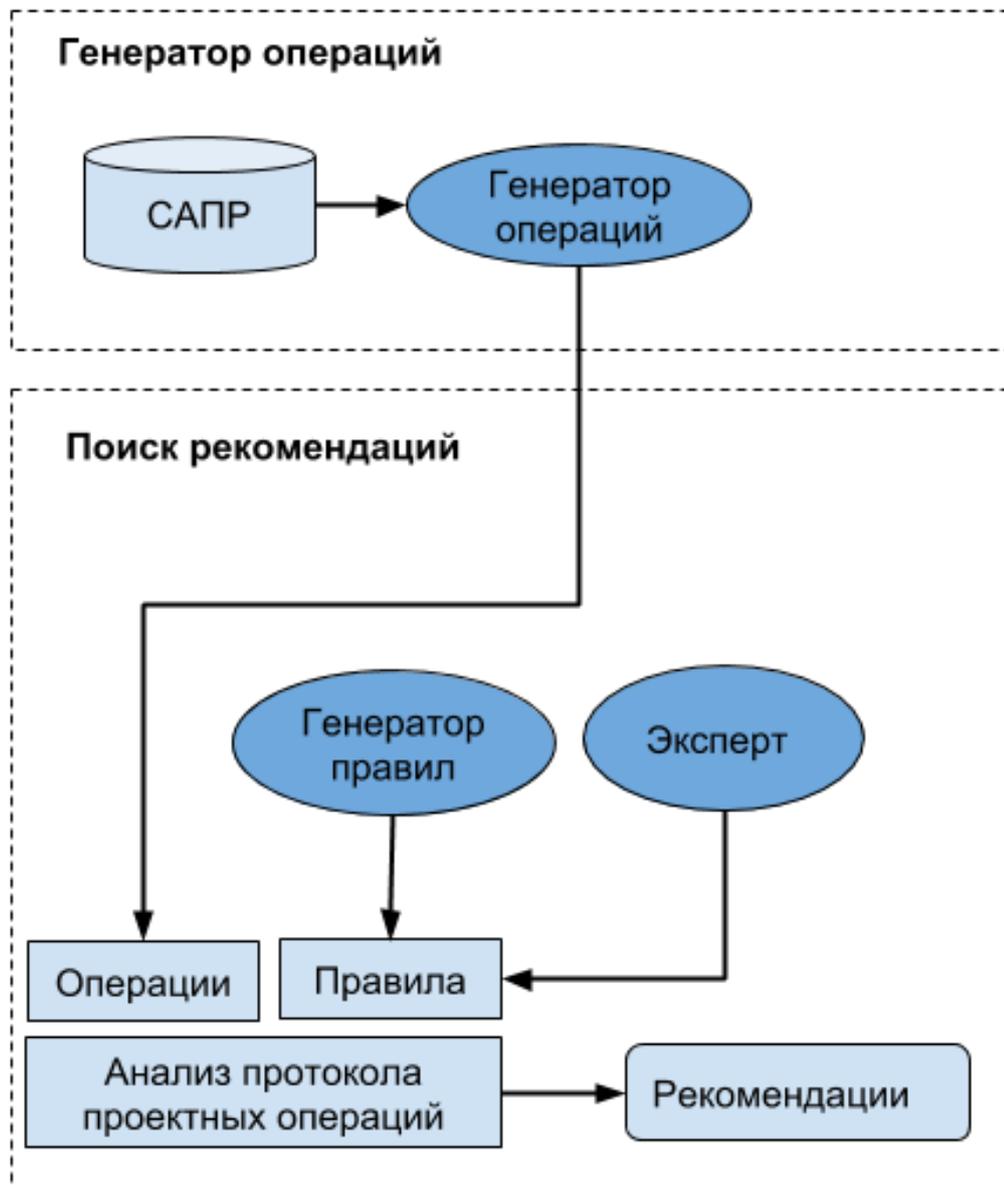


Рисунок 4.11 - Система рекомендаций

Таблица 4.2 - Компоненты генератора операций

Компонент	Описание
KompasTreeManager.cs	Подключение к САПР КОМПАС
ListNode.cs	Список проектных операций
Node.cs	Проектная операция
ObjectNotify.cs	Обработка событий
Part.cs	Обработка деталей
Entity\BaseExtrusion.cs	Операция выдавливания
Entity\BaseRotated.cs	Операция вращения
Entity\BossEvolution.cs	Операция приклеить кинематически

Компонент	Описание
Entity\BossExtrusion.cs	Операция приклеенного выдавливания
Entity\BossRotated.cs	Операция приклеенного вращения
Entity\Chamfer.cs	Операция фаска
Entity\CircularCopy.cs	Операция копирования по окружности
Entity\CircularPartArray.cs	Операция копирования по концентрической сетке
Entity\CutExtrusion.cs	Операция вырезанного выдавливания
Entity\CutRotated.cs	Операция вырезанного вращения
Entity\CylindricSpiral.cs	Операция спираль цилиндрическая
Entity\Entity.cs	Интерфейс элемента модели (оси, плоскости, формообразующего элемента)
Entity\EntityManager.cs	Менеджер интерфейсов элемента модели
Entity\Fillet.cs	Операция скругления
Entity\Hole.cs	Операция отверстие
Entity\MirrorCopy.cs	Операция зеркальный массив
Entity\RDefinition.cs	Операция ребро жесткости
Entity\Sketch.cs	Эскиз
Mate\Mate.cs	Операция сопряжения
Mate\MateAngle.cs	Операция сопряжения под углом
Mate\MateCoincidence.cs	Операция сопряжения совпадение
Mate\MateConcentric.cs	Операция сопряжения соосность
Mate\MateConstraintGroup.cs	Массив сопряжений
Mate\MateDistance.cs	Операция сопряжения «на расстоянии»
Mate\MateInPlace.cs	Операция сопряжения «на месте»
Mate\MateParallel.cs	Операция сопряжения «параллельность»
Mate\MatePerpendicular.cs	Операция сопряжения «перпендикулярность»
Mate\MateTangency.cs	Операция сопряжения «касание»
Shapes\Arc.cs	Обработка дуг

Компонент	Описание
Shapes\Circle.cs	Обработка окружностей
Shapes\LineSegment.cs	Обработка отрезков
Shapes\Point.cs	Обработка точек
Shapes\Shape.cs	Обработка геометрических фигур

Проектные операции кодируются в формате XML [143], пример представления показан ниже:

```
<op type="Отрезок" toptype="shape" id="100549" name="Отрезок">
```

```
<params>
```

```
<param>
```

```
<name>sketch</name>
```

```
<val>100548</val>
```

```
</param>
```

```
<param>
```

```
<name>Начальная точка</name>
```

```
<val>0,00; -3,50. </val>
```

```
</param>
```

```
<param>
```

```
<name>Конечная точка</name>
```

```
<val>0,00; 3,50. </val>
```

```
</param>
```

```
</params>
```

```
</op>
```

```
<op type="Дуга" toptype="shape" id="100552" name="Дуга">
```

```
<params>
```

```
<param>
```

```
<name>sketch</name>
```

```
<val>100548</val>
```

```
</param>
```

```
<param>
```

```

    <name>Центр</name>
    <val>0,00; 0,00</val>
</param>
<param>
    <name>Радиус</name>
    <val>3,50</val>
</param>
<param>
    <name>Начальный угол дуги</name>
    <val>90,00</val>
</param>
<param>
    <name>Конечный угол дуги</name>
    <val>270,00</val>
</param>
</params>
</op>

```

Поиск рекомендаций

Поиск рекомендаций выполнен на платформе Ruby [86], что позволяет быстро вносить новые правила для поиска рекомендаций. Компоненты подсистемы поиска рекомендаций приведены в таблице 4.3.

Таблица 4.3 - Компоненты подсистемы поиска рекомендаций

Компонент	Параметр	Описание
Node		Операций
	Id	Идентификатор операции
	Params	Параметры
	Type	Тип
	Name	Название
Recomend		Рекомендация
	Text	Описание

Компонент	Параметр	Описание
	Ideal_action_size	Количество действий после применения рекомендации
	Current_action_size	Количество действий до применения рекомендации
PartResult		Деталь
	Name	Имя
	Nodes	Операции
	Recomends	Список рекомендаций
	Ideal_action_size	Количество действий после применения рекомендаций
	Current_action_size	Количество действий до применения рекомендаций
Rule		Правило для срабатывания рекомендации
	Name	Имя
	Text	Описание
	Action	Правило срабатывания
RuleEngine		Машина вывода
	Rules	Список правил

Примеры правил для формирования рекомендаций представлены на рисунке 4.12.

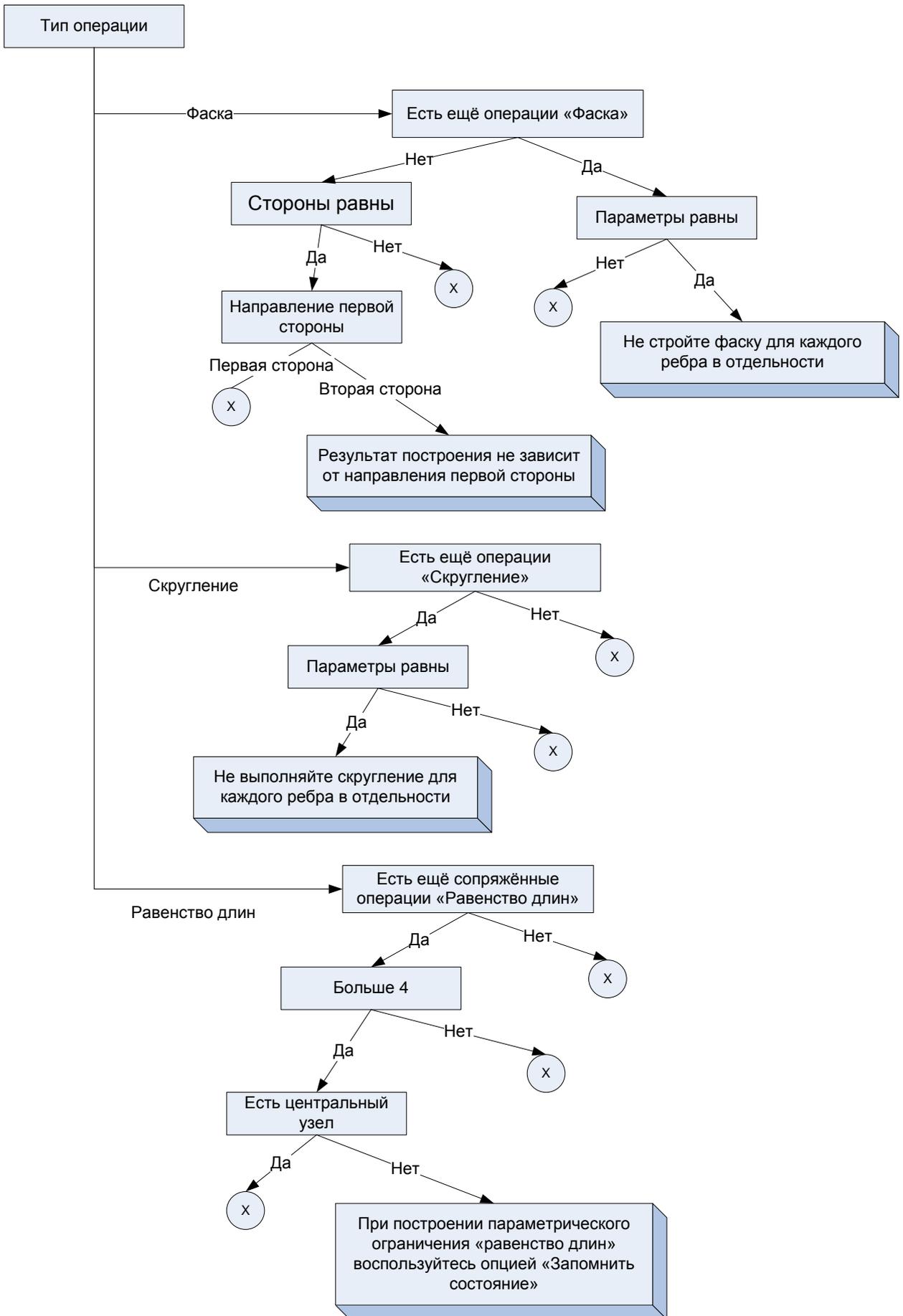


Рисунок 4.12 - Пример дерева правил для формирования рекомендаций

Примеры вывода рекомендаций даны ниже.

1. У вас 2 одинаковых операции («Скругление:3», «Скругление:4»). При построении скругления для множества рёбер с одинаковыми параметрами операции, постарайтесь выбрать как можно большее количество рёбер, это уменьшит количество действий на 38%.

2. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 8 геометрических объектов эскиза «Эскиз:11» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 47%.

3. При построении параметрического ограничения «равенство длин двух отрезков» для 4 геометрических объектов эскиза «Эскиз:13» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 25%.

4.5. Оценка метода формирования персонифицированного сценария обучения

Целью данного подраздела является разработка формульной зависимости в которых рассчитывается время обучения от надёжности и длительности тестов, времени изучения атомов знаний и времени поиска материала в случае дополнительного изучения, степени покрытия тестами атомов знаний, степени владения знанием до обучения, наличия навыков для выполнения практического задания. При этом основными факторами являются:

- исключения учебных материалов на основе тестовых заданий.
- исключения каких-либо этапов сценария на основе проектно-практического опыта.

Пусть сценарий обучения состоит из нескольких этапов, этап сценария включает множество учебных материалов и практическое задание, время необходимое на прохождение этапа:

$$T_{base} = \sum_{i=0}^N t_i + TP,$$

где N - количество учебных материалов,

t_i – время изучения i -го учебного материала,

TP – время выполнения практического задания.

Цель изучения учебных материалов состоит в получении знаний необходимых для выполнения задания. В учебном материале изучается множество атомов знаний, таким образом время необходимое на изучения материала:

$$t_i = \sum_{n=0}^{|ZD_i|} tz_{i,n},$$

где ZD_i - множество атомов знаний, изучаемых в i -ом учебном материале,

$tz_{i,n}$ – время изучения n -го атома в i -ом учебном материале, при этом

$$tz_{i,n} = etz_{i,n} * R_{i,n},$$

где $etz_{i,n}$ – время необходимое для полного изучения n -го атома в i -ом учебном материале,

$R_{i,n}$ – флаг, изучал ли проектировщик n -ый атом в i -ом учебном материале.

Рассмотрим выполнения практического задания как процесс состоящий из двух этапов: поиска дополнительной информации (время изучения неизученных атомов знаний) и непосредственного решения проектного задания. Рассчитаем время выполнения практического задания:

$$TP = \sum_{i=0}^{|ZP|} tdz_i + TP_0,$$

где ZP – множество атомов знаний, необходимых для выполнения практического задания,

tdz_i – время дополнительного изучения i -го атома, $tdz_i = (tzip_i + F_i)(1 - z_i)$,

где $tzip_i$ – время непосредственного изучения i -го атома, F_i – время поиска информации по i -му атому, zip_i – флаг, владения i -ым атомом знания,

TP_0 – время выполнения практического задания при владении всеми необходимыми знаниями.

Введём множество $Z = \bigcup_{i=0}^N ZD_i \cup ZP$ – множество всех атомов знаний, задействованных на данном этапе обучения. Предположим, что время полного изучения конкретного атома не зависит от учебного материала, тогда время обучения будет равно:

$$T_{base} = \sum_{i=0}^N \sum_{j=0}^{|Z|} (tz_j * E_{i,j} * R_{i,j}) + \sum_{i=0}^{|Z|} P_i(tz_i + F_i)(1 - z_i) + TP_0,$$

$$T_{base} = \sum_{j=0}^{|Z|} \left[tz_j * \sum_{i=0}^N (E_{i,j} * R_{i,j}) + P_j(tz_j + F_j)(1 - z_j) \right] + TP_0,$$

$$T_{base} = \sum_{j=0}^{|Z|} tz_j * \left[\sum_{i=0}^N (E_{i,j} * R_{i,j}) + P_j \left(1 + \frac{F_j}{tz_j} \right) (1 - z_j) \right] + TP_0,$$

где tz_j – время необходимое для полного изучения j -го атома,

$E_{i,j}$ – содержится ли j -ый атом в i -ом учебном материале,

P_j – необходимо ли знание j -ого атома для выполнения практического задания.

Для владения атомом знания необходимо или владеть им до обучения, или овладеть в процессе изучения, таким образом $z_i = z_j^0 + (1 - z_j^0) * (1 - \prod_{i=0}^N (1 - E_{i,j} * R_{i,j}))$, где z_i^0 – владение знанием z_i до обучения.

Рассмотрим некоторые механизмы адаптации авторского сценария обучения.

Исключение учебных материалов на основе тестовых заданий.

Тестовая подсистема пытается определить, владеет ли проектировщик знанием z_i . После проведения тестирования имеется следующая информация: rz_i – результат тестирования знания z_i , frz_i – соответствует ли результат тестирования действительному знанию проектировщика, trz_i – время тестирования знания z_i . Показ учебного материала S_i будем в случае хотя бы одного нулевого результата тестирования:

$$S_i = 1 - \prod_{j=0}^{|Z|} [rz_j * MR_j + (1 - E_{i,j})(1 - rz_j * MR_j)],$$

$$rz_i = (1 - z_i^0)(1 - frz_i) + z_i^0 * frz_i,$$

где MR_j – условие тестирования j -го атома.

В этом случае время прохождения этапа сценария будет:

$$T = \sum_{j=0}^{|Z|} tz_j * \left[\sum_{i=0}^N (E_{i,j} * R_{i,j} * S_i) + \frac{trz_j}{tz_j} * MR_J + P_j \left(1 + \frac{F_j}{tz_j} \right) (1 - z_j) \right] + TP_0,$$

владение атомом знания примет вид:

$$z'_i = z_j^0 + (1 - z_j^0) * \left(1 - \prod_{i=0}^N (1 - E_{i,j} * R_{i,j} * S_i) \right).$$

Исключение этапа на основе проектно-практического опыта.

Если у обучаемого проектировщика уже есть навыки для выполнения практического задания, то данный этап сценария исключается из обучения. Наличие навыков определяется путём анализа его ранее выполненных проектных решений. В таком случае, время на прохождение этапа равно 0. В общем виде, время прохождения этапа:

$$T = (1 - EXP) * \left(\sum_{j=0}^{|Z|} tz_j * \left[\sum_{i=0}^N (E_{i,j} * R_{i,j} * S_i) + \frac{trz_j}{tz_j} * MR_J + P_j \left(1 + \frac{F_j}{tz_j} \right) (1 - z_j) \right] + TP_0 \right),$$

где EXP – наличие навыков для выполнения практического задания.

Рассмотрим этап сценария, в котором атомы знаний изучаются только в одном учебном материале, и изучаемых атомов достаточно для выполнения практического задания. Также зафиксируем время изучения атома, время его поиска в случае дополнительного изучения и время тестирования атома знания, тогда:

$$T_{base} = \sum_{j=0}^{|Z|} tz_0 * \left[R'_j + \left(1 + \frac{F_0}{tz_0} \right) (1 - z_j) \right] + TP_0,$$

$$z_j = z_j^0 + R'_j (1 - z_j^0),$$

$$T = (1 - EXP) * \left(\sum_{j=0}^{|Z|} tz_0 * \left[SZ_j * R'_j + \frac{trz_0}{tz_0} * MR_J + \left(1 + \frac{F_0}{tz_0} \right) (1 - z'_j) \right] + TP_0 \right),$$

$$z'_j = 1 - (1 - R'_j * SZ_j) (1 - z_j^0),$$

$$z'_j = z_j^0 + frz_j * R'_j * SZ_j,$$

$$R'_j = \sum_{i=0}^N R_{i,j}, SZ_j = \sum_{i=0}^N (S_i * E_{i,j}),$$

$$SZ_j = 1 - MR_j * rz_j,$$

где R'_j – флаг, изучал ли проектировщик j -ый атом,

SZ_j – показывался ли проектировщику учебный материал с j -ым атомом.

Рассчитаем сокращение времени обучения по формуле:

$$1 - \frac{(1 - EXP) * \sum_{j=0}^{|Z|} \left[SZ_j * R'_j + \frac{trz_0}{tz_0} * MR_j + \left(1 + \frac{F_0}{tz_0} \right) (1 - z'_j) \right]}{\sum_{j=0}^{|Z|} R'_j + \left(1 + \frac{F_0}{tz_0} \right) (1 - z_j)}$$

Пример следующие значения: среднее значение надёжности теста – 0,9[128], длительность тестов от времени изучения $\left(\frac{trz_0}{tz_0}\right)$ – 5%, время поиска в случае дополнительного изучения атома от времени изучения атома $\left(\frac{F_0}{tz_0}\right)$ – 100%. Вероятность изучения проектировщиком атомов знаний при первичном просмотре материала варьируем от 0,7 до 0,9, степень покрытия тестами атомов – 0,7, степень владения знанием до обучения – от 0 до 0,8, вероятность наличия навыков для выполнения практического задания – от 0 до 0,1. Расчёт приведён в таблице 4.4.

Таблица 4.4 - Сокращение времени обучения при использовании персонализированного сценария обучения

№	Вероятность наличия навыков для выполнения практического задания	Вероятность изучения атомов знаний при первичном просмотре материала	Степень владения знаниями до обучения	Относительная длительность обучения	Степень владения знаниями после обучения	Относительная длительность выполнения практического задания	Относительная длительность тестирования	Относительная длительность авторского сценария	Относительная длительность обычного этапа сценария	Сокращение времени обучения
1	0	0,7	0	0,7	0,7	0,7	0,04	1,4	1,3	-6,5%
2	0	0,7	0,4	0,5	0,8	0,4	0,04	0,9	1,1	14,3%
3	0	0,7	0,8	0,3	1,0	0,1	0,04	0,4	0,8	47,1%
4	0	0,9	0	0,8	0,8	0,3	0,04	1,2	1,1	-8,9%

№	Вероятность наличия навыков для выполнения практического задания	Вероятность изучения атомов знаний при первичном просмотре материала	Степень владения знаниями до обучения	Относительная длительность обучения	Степень владения знаниями после обучения	Относительная длительность выполнения практического задания	Относительная длительность тестирования	Относительная длительность авторского этапа сценария	Относительная длительность обычного этапа сценария	Сокращение времени обучения
5	0	0,9	0,4	0,6	0,9	0,1	0,04	0,8	1,0	20,0%
6	0	0,9	0,8	0,4	1,0	0,0	0,04	0,4	0,9	53,9%
7	0,1	0,7	0	0,7	0,7	0,7	0,04	1,2	1,3	4,2%
8	0,1	0,7	0,4	0,5	0,8	0,4	0,04	0,8	1,1	22,8%
9	0,1	0,7	0,8	0,3	1,0	0,1	0,04	0,4	0,8	52,4%
10	0,1	0,9	0	0,8	0,8	0,3	0,04	1,1	1,1	2,0%
11	0,1	0,9	0,2	0,7	0,9	0,2	0,04	0,9	1,1	14,5%
12	0,1	0,9	0,4	0,6	0,9	0,1	0,04	0,7	1,0	28,0%
13	0,1	0,9	0,8	0,4	1,0	0,0	0,04	0,4	0,9	58,5%

В среднем сокращение времени обучения составляет 24%.

Конструкторам соответствуют такие строчки 8, 9, 12, 13 в таблице 4.4, так как у них есть навыки для выполнения практического задания, и высокая степень владения знанием до обучения.

Также для оценки эффективности применения персонифицированного сценария обучения были сформированы целевые группы студентов. Характеристики эксперимента: длительность исследования 3,5 часа, вовлечено 9 студентов, которые были разделены на 2 целевые группы численностью по 4-5 человек, цель обучения – формирование компетенций: умеет работать с интерфейсом; владеет навыками построения эскизов; владеет навыками твердотельного моделирования. Характеристики групп примерно соответствуют 11 строке в таблице 4.4 (вероятность наличия навыков для выполнения практического задания – 0,1, вероятность изучения атомов знаний при первичном просмотре материала – 0,9, степень владения знанием до обучения – 0,2). В одной группе в процессе обучения применялся метод формирования персонифицированного сценария обучения, в другой группе применялись только

стандартные средства обучения. В среднем время обучения у студентов, использующих персонифицированный сценарий обучения на 14% меньше, а время выполнения контрольного задания на 4%, чем у остальных студентов. Первичные данные приведены в таблице 4.5.

Таблица 4.5 - Экспериментальные данные для оценки метода формирования персонифицированного сценария обучения

Студент	Начало	Этап 1	Этап 2	...	Этап 9	Контрольное задание	Время обучения, мин	Время выполнения контрольного задания, мин
Авторский сценарий обучения								
Горшков	9:15	9:50	9:50	...	11:10	12:05	115	55
Матвеев	9:15	9:58	10:14	...	11:10	12:06	115	56
Соловьев	9:20	9:53	10:18	...	11:40	12:18	140	38
Линьков	9:25	9:48	10:20	...	11:28	12:07	123	39
Сизов	10:06	-	-	...	10:54	12:08	48	74
Среднее							108	52
Обычный сценарий обучения								
Филиферов	9:15	9:55	10:14	...	11:44	12:40	149	56
Захарычев	9:15	9:53	10:14	...	11:28	12:32	133	64
Еланцев	9:15	9:58	10:17	...	11:42	12:30	147	48
Долгов	10:20	10:33	10:37	...	11:37	12:28	77	51
Среднее							127	55

4.6. Оценка повышения эффективности деятельности проектировщика при использовании системы рекомендаций

В качестве меры эффективности деятельности проектировщика выберем количество действий, выполняемых при построении детали.

Для оценки метода формирования рекомендаций были проанализированы сборки, разработанные в САПР КОМПАС. Источниками сборок являются работы, выполненные студентами Ульяновского Государственного Технического

Университета, специалистами АО «Ульяновский механический завод», а также сеть Интернет. Результаты представлены в таблице 4.6.

Таблица 4.6 - Результаты анализа сборок

Сборка	Количество операций	Количество действий	Количество рекомендаций	Относительное значение уменьшения количества действий, %
Насос	830	3578	16	3,0%
Редуктор 2ЧМ-40-40-51- Ц-У2	220	913	2	1,1%
Блок направляющий	318	1330	7	4,0%
Подвеска	937	3967	6	1,1%
Ступица в сборе	162	742	4	1,0%
Сборка1	46	212	3	20,8%
Сборка2	123	542	1	3,1%
Сборка3	58	287	1	5,9%
Сборка4	119	552	3	3,4%
Сборка5	1425	6075	6	0,9%
ИТОГО	4238	18198	49	4,4%

Таким образом, разработанная система позволяет уменьшить количество действий, выполняемых проектировщиком в среднем на 4,4%.

Предложенные рекомендации обсуждались с преподавателями Машиностроительного факультета Ульяновского Государственного Технического Университета, конструкторами АО «Ульяновский механический завод», которые подтверждают корректность сформированных рекомендаций и востребованность предложенной системы.

Оценим работу системы рекомендаций при построении сборки «Насос» (рисунок 4.13) состоящей из 21 детали. Подробный вывод подсистемы рекомендаций приведён в приложении 4. Сводные данные по всем деталям приведены в таблице 4.7.

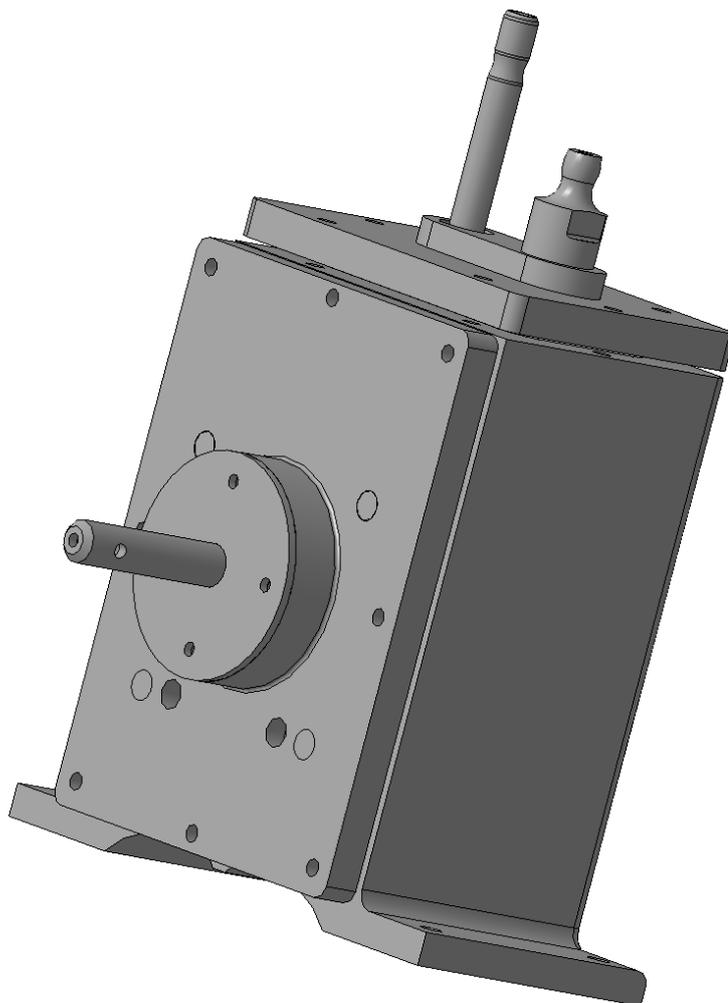


Рисунок 4.13 - Сборка «Насос»

Таблица 4.7 - Результаты использования рекомендаций для сборки «Насос»

Деталь	Количество операций	Количество действий	Количество рекомендаций	Уменьшение количества действий	
Вакуумная камера	111	487	2	7	1.4%
Держатель клапана	13	52	0	0	0.0%

Деталь	Количество операций	Количество действий	Количество рекомендаций	Уменьшение количества действий	
Иллюминатор	23	101	0	0	0.0%
Клапан	12	47	0	0	0.0%
Колесо ременной передачи	81	330	1	2	0.6%
Корпус насоса	249	1044	4	19	1.8%
Крышка вакуумной камеры	44	195	1	5	2.6%
Крышка верхняя	50	218	0	0	0.0%
Крышка подшипника	19	80	1	2	2.5%
Лопасть	13	59	0	0	0.0%
Передняя крышка	127	548	5	17	3.1%
Подшипник скольжения	22	96	1	2	2.1%
Прокладка верхняя	37	151	1	5	3.3%
Прокладка передняя	44	182	1	8	4.4%
Прокладка под выводящую трубку	5	21	0	0	0.0%
Пружина	7	27	0	0	0.0%
Ротор	51	222	0	0	0.0%
Трубка входная	37	146	0	0	0.0%
Выводящая трубка	48	207	0	0	0.0%
Шпилька	54	211	0	0	0.0%
Штифт	22	83	0	0	0.0%
Сборка «Насос»	1069	4507	17	67	1.5%

Таким образом, сокращение действий необходимых для построения сборки «Насос» при использовании рекомендаций составляет 1.5%.

4.7. Реализация виртуальных компонент для системы формирования профиля компетенций проектировщика

Тренажёры для СФПК написаны на языке программирования JavaScript [106], на основе автоматного подхода [57] и прототипно-ориентированного программирования [126]. Активно использовались следующие библиотеки: jQuery [34], flotcharts [6] и Raphaël [50, 96].

Программный комплекс тренажёра состоит из ядра, контроллеров (логических и внешних), рендеринга, менеджера пользовательских событий и расширения ядра. Структура компонентов показана рисунке 4.14.

Ядро тренажёра несёт небольшую функциональную значимость, основная его работа - это маршрутизация событий от контроллеров. Оно позволяет подписываться на события и отправлять их. Также ядро отвечает за режимы работы тренажёра, за его остановку и продолжение работы. Ядро не зависит от конкретного тренажёра.

Основную работу выполняют контроллеры, они получают сигналы от пользователей и от ядра, изменяют состояния тренажёра, отвечают за внешний вид. Различают два вида контроллеров: логические и внешние. Логические тренажёры не имеют внешнего поведения, то есть пользователь с ними непосредственно не взаимодействует. Следовательно, данный компонент не представлен на панели тренажёра. Например, менеджер ошибок, слушает событие «Ошибка входного сигнала» и при необходимости выдает сообщение, что пользователь ввёл некорректные данные. Внешние контроллеры, напротив, взаимодействуют с пользователем и представлены на панели тренажёра. Такой контроллер может обрабатывать событие «action», взаимодействие с пользователем, например, щелчок по кнопке, движение ползунка и так далее.

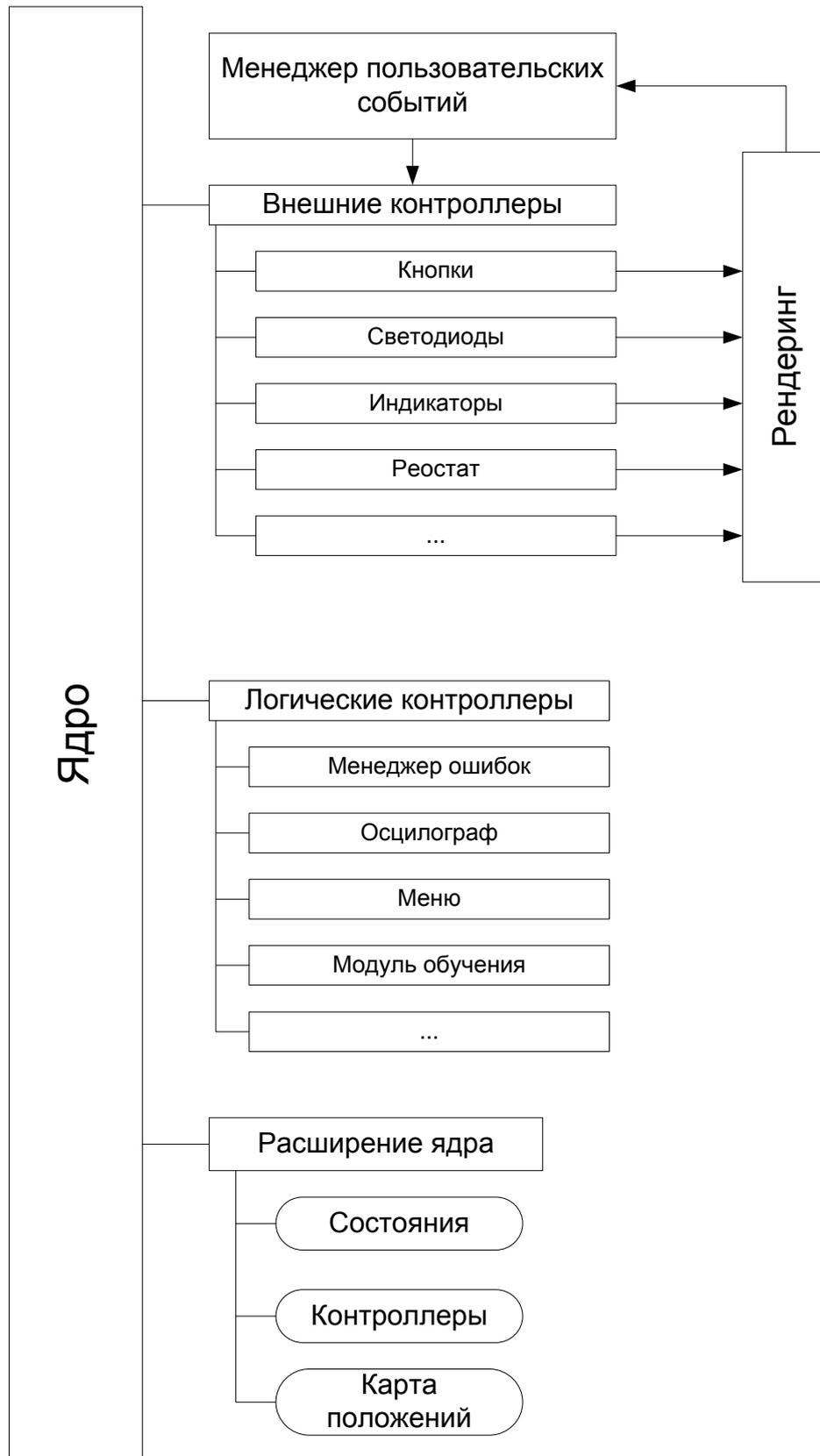


Рисунок 4.14 - Структура тренажёра [100]

Имеются внешние контроллеры, которые не получают таких событий, например, индикаторы, они только выводят сообщения и непосредственно не реагируют на пользователя. Каждый экземпляр контроллера имеет уникальный идентификатор, благодаря которому к нему можно получить доступ.

Работа внешних контроллеров невозможна без рендеринга, вместе они составляют одно целое. Рендеринг выводит пользователю состояние контроллера, подсветку кнопки, включение светоиндикатора, выводит на экран данный компонент.

Менеджер пользовательских событий слушает действия, выполняемые пользователями, и отправляет их конкретному внешнему контроллеру, в зоне действия которого они произошли для обработки.

Расширение ядра – это часть, которая уникальна для данного тренажёра. Каждый конкретный тренажёр описывается состоянием, которое полностью определяет тренажёр и список контроллеров, определяющих его поведение. Сами контроллеры различаются параметрами, например, параметры кнопки, определяют, каким образом будет изменено состояние тренажёра. Карта положений определяет местоположение внешних контроллеров на тренажёре, задаётся массивом.

Пример функционирования тренажёра

Рассмотрим функционирование тренажёра, состоящего из индикатора и двух кнопок, включающих и выключающих индикатор (рисунок 4.15). Состояние тренажёра описывается одним параметром «indicatorOn», если оно равно единице, то индикатор включён, если нет – выключен. Дадим описание двум контроллерам: индикатор и кнопка.

Контроллер «Индикатор-светодиод» подписывается через ядро на событие «state_change» (изменение состояния тренажёра), при возникновении данного события контроллер в соответствии со своими параметрами определяет, должен ли сейчас он гореть или нет. Для включения он вызывает функцию рендера «lightOn» (включить), для выключения – «lightOff» (выключить). Параметром индикатора является функция, возвращающая единицу, когда он должен гореть, и ноль – когда не должен. Для данного тренажёра параметром является «state.indicatorOn».

Контроллер «Кнопка» подписывается на событие «action», которое происходит при взаимодействии с данным контроллером. Параметром кнопки является функция смены состояния тренажёра. Для кнопки включения индикаторы - это «state.indicatorOn = 1», для кнопки выключения – «state.indicatorOn = 0». После смены состояния кнопка отсылает событие «state_change» (изменение состояния тренажёра).

Рассмотрим процесс включения индикатора:

- 1) пользователь нажимает на кнопку включения индикатора;
- 2) кнопка обрабатывает действие:
 - меняет состояние «indicatorOn» на единицу,
 - отсылает сообщение «state_change» (изменение состояния тренажёра);
- 3) индикатор обрабатывает событие «state_change»:
 - функция «state.indicatorOn» возвращает единицу,
 - индикатор загорается.

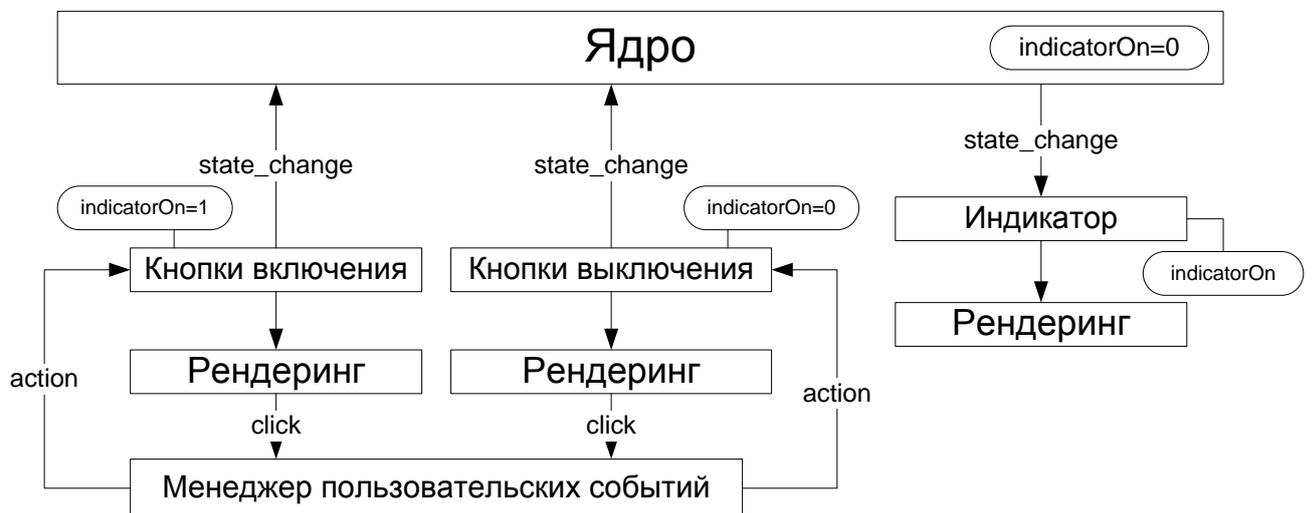


Рисунок 4.15 - Пример тренажёра [100]

Описание контроллеров приведены в приложении 3.

4.8. Реализации ядра мобильной системы обучения, моделей предметной области, обучаемого, тестирования для платформы Android

В качестве одной из платформ реализации СФПК выбрана Android [5], у которой по данным исследовательской компании Strategy Analytics около 81,2% рынка [36].

Разработка архитектуры

Выделим следующие требования к разрабатываемой системе:

- повышение интерактивности и гибкости процесса обучения в сравнении с существующими разработками;
- реализация персонифицированного метода обучения для учета индивидуальных особенностей обучаемого;
- разработка инструментов, автоматизирующих процесс наполнения системы учебным материалом;
- планирование и прогнозирование сценария обучения;
- возможность включения в курс различных мультимедиа объектов;
- простота использования, интуитивно-понятный интерфейс взаимодействия с пользователями;
- хранение, сбор и наглядное представление статистики процесса обучения;
- работа через мобильное оборудование в сетях Интернет и Интранет.

В соответствии с приведённым анализом обучающих систем выделим основные характеристики разрабатываемой системы. За основу выберем клиент-серверную архитектуру. Реализацией клиента является мобильное приложение, посредством которого пользователь системы будет взаимодействовать с обучающей средой. Сервер представляет собой ядро, основной задачей которого является обеспечение связей и функционирования модулей системы. Основными модулями системы являются: представление материала (обеспечивает пересылку информации на клиентское приложение для показа пользователю), пространство обучения (модели пользователя, сценария, предметной области), управления пользователями (авторизация, аутентификация, роли и права, профиль

пользователя), тестирование, система отчётов. Основные расчёты производятся на сервере различными модулями, в целях минимизации трафика на стороне клиента расположен кэш данных [91] (профиль пользователя и учебный материал). Разработанная архитектура представлена на рисунке 4.16.

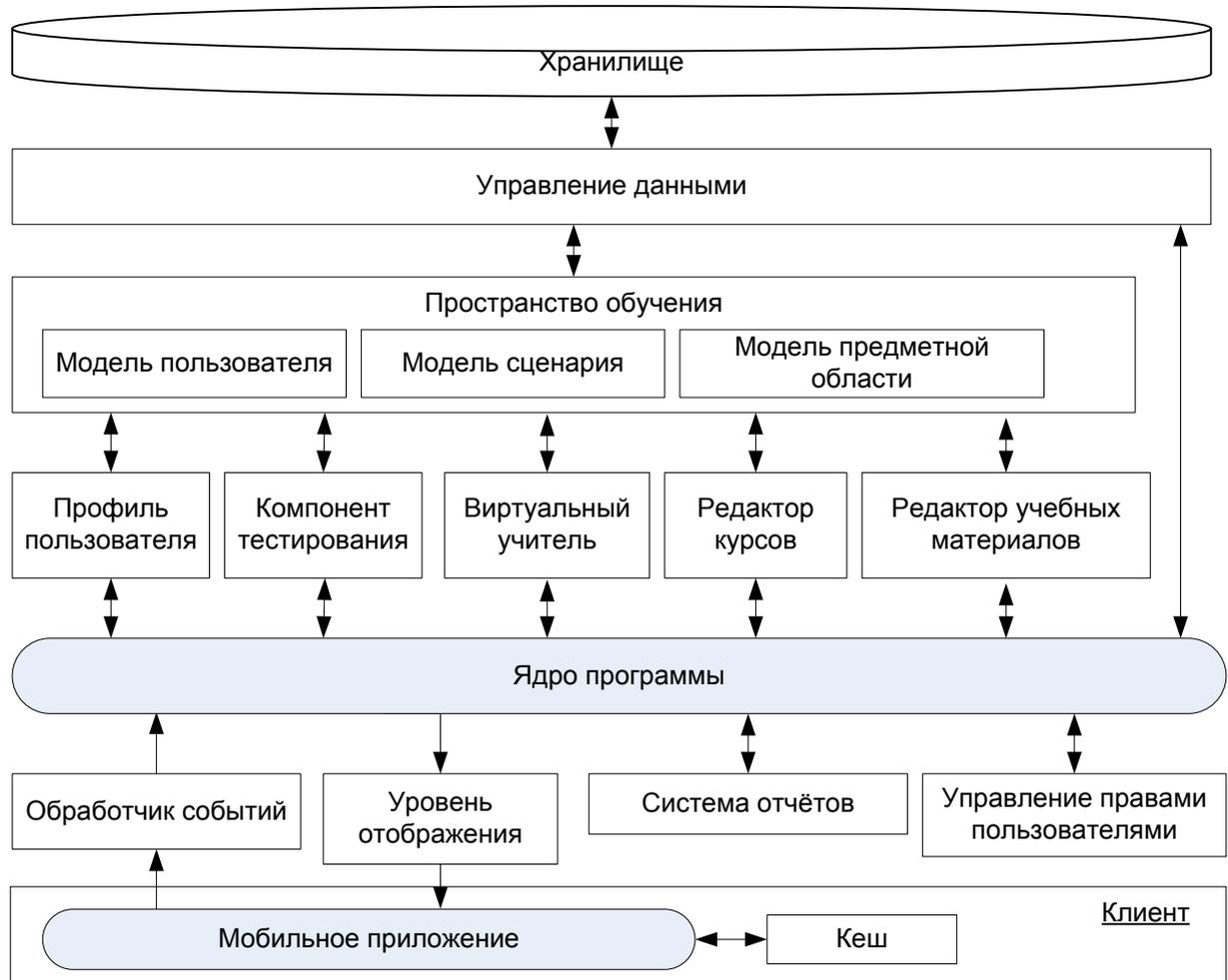


Рисунок 4.16 - Архитектура интеллектуальной обучающей системы на базе мобильных технологий [15]

На первом этапе были реализованы ядро обучающей системы и модели пространства обучения (модель предметной области, модель обучаемого, модель тестирования) [60, 69].

Представим общее описание части обучающей системы (работа с предметной областью, профилем обучаемого и тестами) в виде диаграмм вариантов использования (рисунок 4.17) и деятельности (рисунок 4.18).

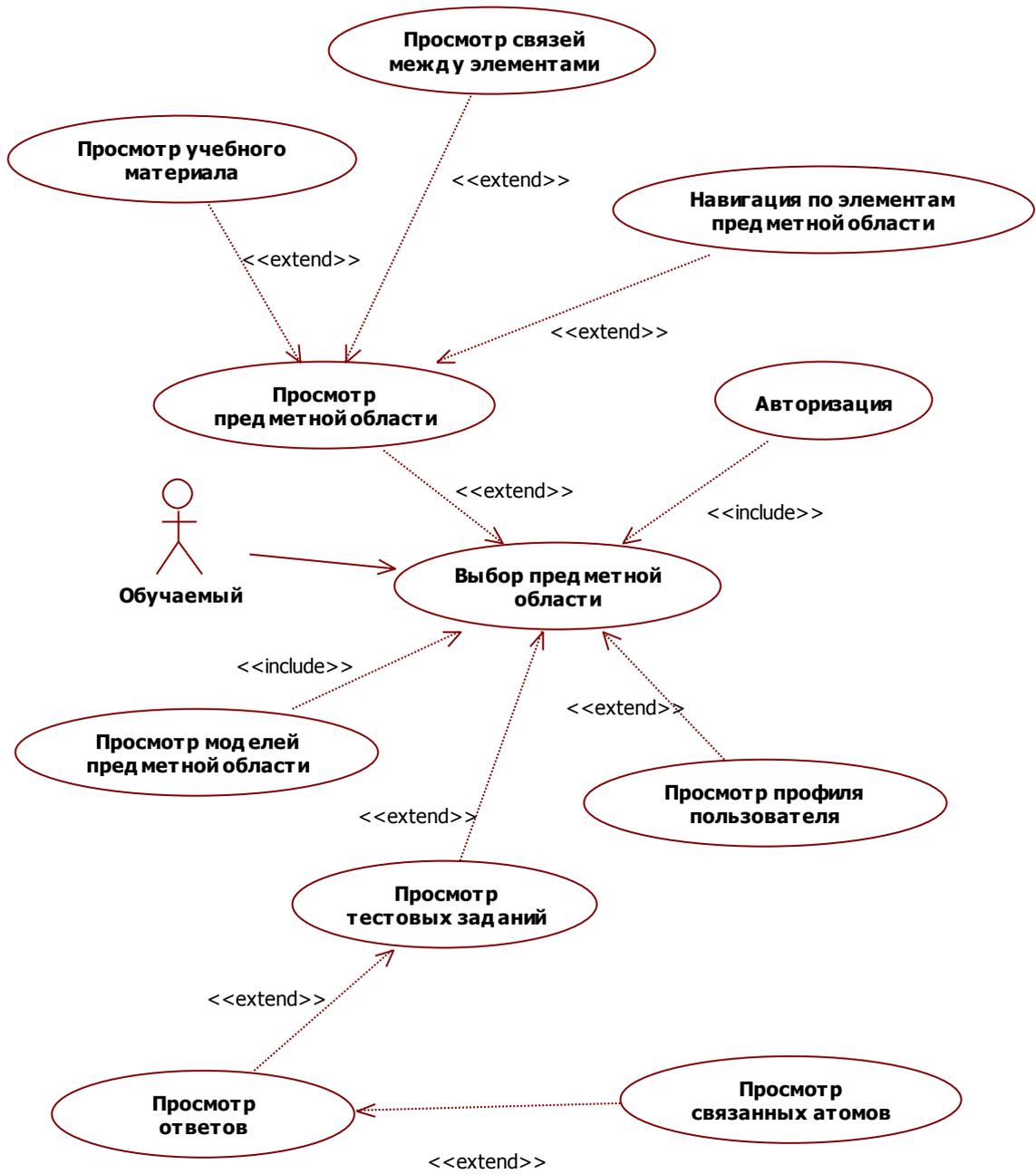


Рисунок 4.17 - Варианты использования разработанных компонентов мобильной обучающей системы

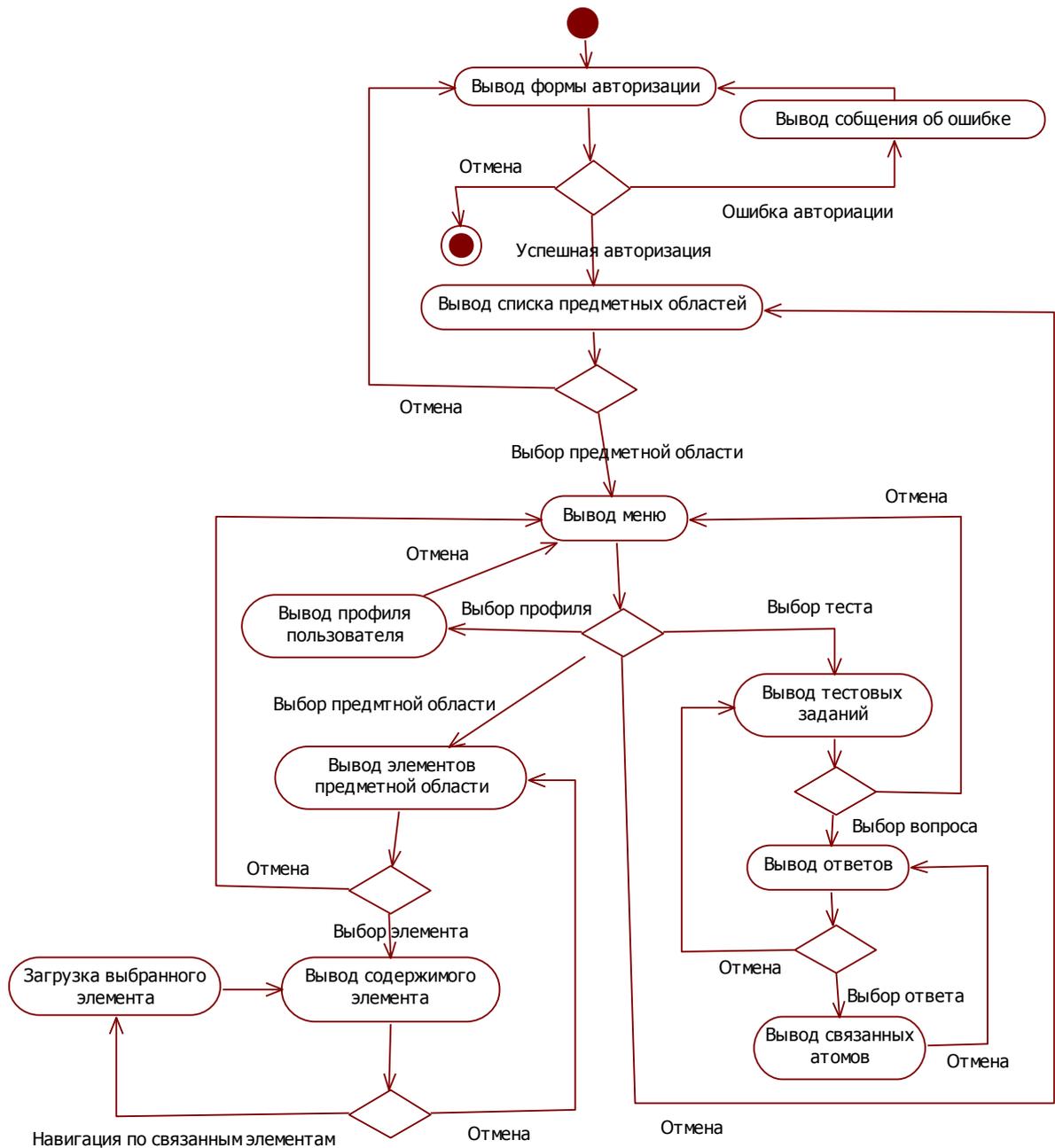


Рисунок 4.18 - Диаграмма деятельности мобильной обучающей системы

Следующим этапом разработаем ER-диаграмму базы данных (рисунок 4.19). Основные концепции модели «сущность-связь» включают типы сущностей, типы связей и атрибуты [105]. Объектные отношения хранят данные обо всех объектах одного и того же типа, по одному кортежу на объект, и содержат ключ для идентификации объекта. Все другие поля должны функционально зависеть от этого ключа.

Поставленная задача предполагает наличие восьми сущностей.

1. Пользователь (users).
2. Модель предметной области (model).
3. Элемент (node).
4. Связь между элементами (aggregationrelation).
5. Модель обучаемого (student).
6. Тестовое задание (test).
7. Вариант ответа (test_answer).
8. Связанный атом предметной области с ответом на тестовое задание (test_answer_node).

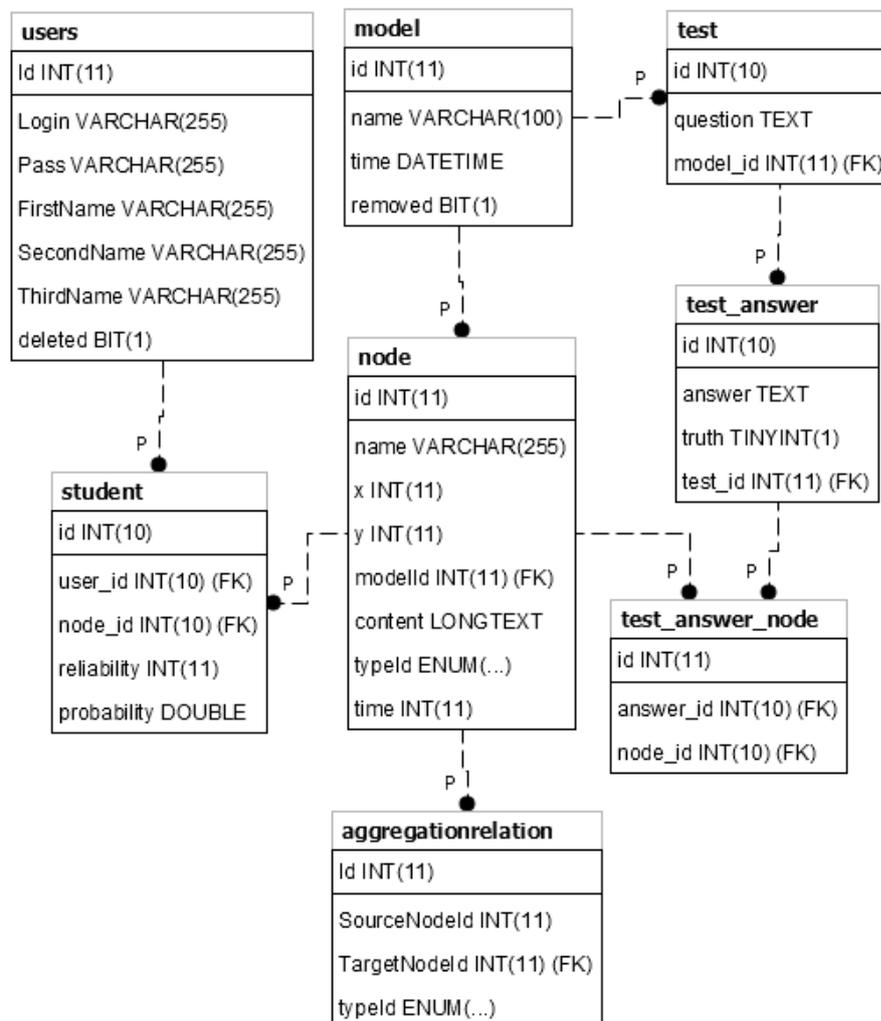


Рисунок 4.19 - ER-модель БД

Сущность пользователь (users) – хранит список пользователей, допущенных к использованию обучающей системы (таблица 4.8). Авторизация происходит на основе логина и пароля.

Таблица 4.8 - Описание сущности - пользователь (users)

Атрибуты	Описание
id	Идентификатор
Login	Логин
Pass	Пароль
FirstName	Имя
SecondName	Фамилия
ThirdName	Отчество

Сущность модель предметной области (model) - хранит названия и даты изменения всех созданных моделей (таблица 4.9), к модели предметной области также привязываются тестовые задания и профиль обучаемого.

Таблица 4.9 - Описание сущности - модель предметной области (model)

Атрибуты	Описание
id	Идентификатор
name	Название
time	Дата создания
removed	Модель удалена

Сущность элемент(node) - хранит элементы предметной области (таблица 4.10). Выделяются следующие типы элементов:

- атом - непересекающиеся между собой и далее неделимые для целей обучения ячейки знаний;
- понятие – объединение атомов по какому-либо принципу;
- учебный материал – материал для изучения;
- справочный материал - дополнительным материалом для изучения, который присутствует как ссылки для необязательного ознакомления.

Таблица 4.10 - Описание сущности - элемент(node)

Атрибуты	Описание
id	Идентификатор
name	Название
x	Положение по горизонтали
y	Положение по вертикали
modelId	Модель предметной области
content	Содержание
typeId	Тип узла
time	Время изучения материала

Сущность связь между элементами (aggregationrelation) – хранит связи между элементами (таблица 4.11). Связи бывают следующих типов:

- «состоит_в»- объединение атомов\понятия в понятия;
- «изучается_после» - определяет порядок изучения атомов;
- «изучается_в» - определяет задействованные понятия в учебном и справочном материале.

Таблица 4.11 - Описание сущности – связь между элементами (aggregationrelation)

Атрибуты	Описание
Id	Идентификатор
SourceNodeId	Начальный элемент
TargetNodeId	Конечный элемент
modelId	Модель предметной области
typeId	Тип связи

Сущность модель обучаемого (student) – хранит в себе профиль пользователя (таблица 4.12). Модель обучаемого представляет собой оверлейную модель и несёт информацию о степени владения знаниями для каждого атома и количества контрольных измерений.

Таблица 4.12 - Описание сущности – модель обучаемого (student)

Атрибуты	Описание
id	Идентификатор
user_id	Пользователь
node_id	Атом знания
reliability	Количество контрольных измерений
probability	Степень владения знаниями
model_id	Модель предметной области

Сущность тестовое задание (test) – хранит список тестовых заданий для заданной предметной области (таблица 4.13).

Таблица 4.13 - Описание сущности – тестовое задание (test)

Атрибуты	Описание
id	Идентификатор
question	Вопрос
model_id	Модель предметной области

Сущность вариант ответа (test_answer) – хранит список возможных ответов для тестового задания (таблица 4.14).

Таблица 4.14 - Описание сущности – вариант ответа (test_answer)

Атрибуты	Описание
id	Идентификатор
answer	Ответ
truth	Правильный ответ
test_id	Идентификатор вопроса

Сущность связанный атом знания с ответом на тестовое задание (test_answer_node) – хранит список атомов знания, которые связаны с выбранными ответами на тестовое задание (таблица 4.15).

Таблица 4.15 - Описание сущности - связанный элемент предметной области с ответом на тестовое задание (test_answer_node)

Атрибуты	Описание
id	Идентификатор
answer_id	Идентификатор ответа
node_id	Задействованный атом знания

Разработанная система состоит из 24 модулей.

1. AggregationRelation – обработка связей между элементами.
2. EnumRelation – работа с типами связей (группировка, определение, зависимость, представление).
3. Model – работа с предметной областью.
4. ModelNode – работа с элементами предметной области.
5. ModelNodeGeneratorContent – формирование содержимого элемента.
6. ModelNodeStudy – работа с уровнем усвоения предметной области обучаемым.
7. NodeEnum – работа с типами элементов (атом, понятие, учебный материал, справочный материал).
8. SqlValue – работа со значениями из СУБД.
9. Student – профиль пользователя.
10. Test – тестовые задания.
11. TestAnswer – ответы на тестовые задания.
12. User – Работа с пользователями, авторизация.
13. Core – ядро обучающей системы.
14. DBProvider – работа с базой данных.
15. DKListActivity – базовая форма для обработки списков.
16. LoginActivity- форма авторизации (рисунок 4.20).
17. ListModelActivity – форма для выбора предметной области.
18. ModelActivity – форма-меню для выбора определённого раздела (предметная область, профиль обучаемого, тестирование).

19. DomainModelActivity – форма для работы со списком элементов предметной области.

20. ModelNodeActivity – форма для просмотра содержимого элемента предметной области (рисунок 4.21).

21. StudentActivity – форма для просмотра профиля обучаемого;

22. TestActivity – форма для просмотра списка тестовых заданий (рисунок 4.22).

23. QuestionActivity – форма для просмотра ответов на тестовые задания.

24. AnswerNodeActivity – форма для просмотра связанных элементов предметной области с ответом на тестовое задание.

The image shows a login form with a title bar 'Вход'. It contains two input fields: 'Логин' (Login) and 'Пароль' (Password). Below the fields is a button labeled 'Вход' (Login).

Рисунок 4.20 - Форма авторизации

Информация об узле

Урок №13. Вращение(1), 20 мин

Тип: учебный материал

Родители:

Дочерние элементы:

1. [Операция «Вращение»\(4\)](#)

Содержание:

ЗАНЯТИЕ 1.4. КОМПАС-3D

Операции Выдавливания и Вращения

Операция выдавливания

Позволяет создать основание детали, представляющее собой тело выдавливания.

Тело выдавливания образуется путем **перемещения эскиза** в направлении, перпендикулярном его плоскости.

Команда доступна, если выделен один эскиз.

Для вызова команды нажмите кнопку **Операция выдавливания** на **Инструментальной панели трехмерных построений**.

Рисунок 4.21 - Форма для просмотра содержимого элемента предметной области

Тестовые задания

1. Укажите способы построения касательных отрезков?
2. Какое предложение наиболее точно описывает понятие UML об актерах?
3. Какого типа структурных диаграмм НЕТ в UML?
4. Какие способы построения прямых отрезков вы знаете?
5. Что такое окружность?

Рисунок 4.22 - Форма для просмотра списка тестовых заданий

Программная реализация ядра мобильной системы

Ядро мобильной системы отвечает за авторизацию пользователей и работу с сервером базы данных. В качестве СУБД выбрана MySQL 5.1, которая является свободной системой и полностью соответствует целям разработки.

Функциональность ядра обеспечивается за счёт модулей User, DBProvider, Core, LoginActivity, ModelActivity. Обработывает сущность «Пользователь».

Программная реализация модели предметной

Функциональность обеспечивается за счёт модулей Model, ModelNode, AggregationRelation, EnumRelation, ModelNodeGeneratorContent, NodeEnum, ListModelActivity, DomainModelActivity, ModelNodeActivity. Обработывает сущности «Модель предметной области», «Элемент», «Связь между элементами».

Программная реализация модели

Функциональность обеспечивается за счёт модулей Student, ModelNodeStudy, ModelNode. Обработывает сущность «Модель обучаемого».

Программная реализация модели

Функциональность обеспечивается за счёт модулей Test, TestAnswer, ModelNode, TestActivity, QuestionActivity, AnswerNodeActivity. Обработывает сущности «Тестовое задание», «Вариант ответа», «Связанный элемент предметной области с ответом на тестовое задание».

4.9. Выводы и рекомендации

1. Разработанные программные средства формирования профиля проектировщика позволили апробировать на практике предложенные в работе методы и алгоритмы и в результате проведенных экспериментов полностью доказали свою состоятельность.

2. Реализуемая СФПК является расширяемой, то есть поддерживает модульный принцип и позволяет расширять свою функциональность за счёт плагинов. Плагины позволяют расширять как клиентскую, так и серверную часть, благодаря предоставлению соответствующего API.

3. Использование клиент-серверной архитектуры для организации системы формирования профиля проектировщика позволяет проектировщику работать с системой из любого места, в котором доступен интернет.

4. Разработанная архитектура позволила минимизировать требования к аппаратному и программному обеспечению рабочих мест пользователей.

Пользователям необходимо следить за актуальной версией браузера для того, чтобы корректно работал функционал клиентской части (веб-ориентированная система представления учебно-методического материала).

5. Реализация представления учебно-методического материала на платформе Android позволила опробовать на практике предложенные в работе модели, методы и средства. А также пройти обучение и теоретическое тестирование, в том числе и с мобильного телефона.

6. Разработанные компоненты «Генератор протокола проектных операций» и «Система рекомендаций» позволяют сформировать рекомендации и оценить количество выполняемых действий при проектировании в САПР КОМПАС.

7. Для оценки эффективности применения персонифицированного сценария обучения был проведён эксперимент над целевыми группами студентов. В среднем время обучения у студентов, использующих персонифицированный сценарий обучения на 14% меньше, а время выполнения контрольного задания на 4%, чем у остальных студентов. Предложена теоретическая оценка сокращения времени обучения при использовании персонифицированного сценария обучения. В среднем сокращение времени обучения для различных групп обучаемых составляет 24% за счёт исключения учебных материалов на основе тестовых заданий и этапов сценария на основе проектно-практического опыта.

8. Проведён анализ эффективности деятельности проектировщика при использовании системы рекомендаций. Рекомендательная система позволяет сократить количество выполняемых действий проектировщиком при трехмерном моделировании деталей в САПР КОМПАС в среднем на 4,4%

9. Рекомендуются применение разработанных средств для повышения эффективности деятельности проектировщика в системах автоматизированного машиностроительного проектирования.

10. На программные продукты получен ряд свидетельств Роспатента [129, 130, 131, 132, 133, 134, 135].

Заключение

Цель диссертационной работы – повышение эффективности деятельности проектировщика в системах автоматизированного машиностроительного проектирования за счет формирования необходимых компетенций и рекомендаций для проектировщика – достигнута.

Основными результатами работы являются.

1. Предложена онтологическая модель предметной области машиностроительного проектирования, отличающаяся введением классов, атомов и функций, ориентированных на предметную область автоматизированного машиностроительного проектирования.

2. Предложен новый метод формирования персонифицированного сценария обучения, отличающийся использованием динамических механизмов взаимодействия моделей пространства обучения (профиля проектировщика, теста, проектно-практического задания) с онтологической моделью предметной области автоматизированного машиностроительного проектирования, и позволяющий сократить время обучения на 14%.

3. Предложен новый метод формирования рекомендаций и корректировки компетенций проектировщика, на основе протокола операций трехмерного моделирования деталей, выполняемых в САПР КОМПАС. Рекомендательная система позволяет сократить количество выполняемых действий проектировщиком в среднем на 4,4%.

4. Разработана и реализована архитектура автоматизированной обучающей системы с поддержкой персонифицированного обучения, практических задач и формированием рекомендаций в процессе работы проектировщика в САПР КОМПАС.

Список литературы

1. A Wiki-based Environment for Constraint-based Recommender Systems Applied in the E-Government Domain / S. Reiterer, A. Felfernig, M. Jeran et al. // 3rd Workshop on PErsonalization in eGOVernment and Smart Cities: Smart Services for Smart Territories. – 29th June - 3rd July 2015, Dublin. – pp. 1-10.
2. A. Afanasyev, N. Voit, D. Kanev, T. Afanaseva. Organization, development and implementation of intelligent learning environments // 10th International Technology, Education and Development Conference - IATED, 2016. – pp. 2232–2242.
3. A. Khennane. Introduction to Finite Element Analysis Using MATLAB and Abaqus / A. Khennane. – Taylor & Francis, 2013. – p. 487.
4. An Integrated Environment for the Development of Knowledge-based Recommender Applications / Felfernig, G. Friedrich, D. Jannach et al. // International Journal of Electronic Commerce. – 2006. – №11(2). – pp. 11–34.
5. Android для программистов. Создаем приложения / Дейтел П., Дейтел Х., Дейтел Э. и др. - Санкт-Петербург [и др.]: Питер, 2013. - (Библиотека программиста). - 557 с.
6. Attractive JavaScript plotting for jQuery [Электронный ресурс]. — Режим доступа: <http://www.flotcharts.org> (дата обращения: 20.12.2015).
7. Bagheri M. M. Intelligent and Adaptive Tutoring Systems: How to Integrate Learners //International Journal of Education. – 2015. – V. 7. – №. 2. – pp. 1-16.
8. Bener A. B., Ozadali V., Ilhan E. S. Semantic matchmaker with precondition and effect matching using SWRL //Expert Systems with Applications. – 2009. – Т. 36. – №. 5. – pp. 9371-9377.
9. C. Martins. User Modeling in Adaptive Hypermedia Educational Systems / C. Martins, C. Faria, Vaz de Carvalho et al. //Educational Technology & Society. – 2008. – V. 11. – №. 1. – pp. 194-207.
10. Cerami E. Web services essentials: distributed applications with XML-RPC, SOAP, UDDI & WSDL. – O'Reilly Media, 2002.
11. Chen L., Pu P. Critiquing-based recommenders: survey and emerging trends //User Modeling and User-Adapted Interaction. – 2012. – Т. 22. – №. 1-2. – pp. 125-150.

12. Cheryl Lemke. Intelligent Adaptive Learning: An Essential Element of 21st Century Teaching and Learning [Электронный ресурс] – Режим доступа: <http://www-static.dreambox.com/wp-content/uploads/2013/03/white-paper-intelligent-adaptive-learning-21st-century-teaching-and-learning.pdf> (дата обращения: 20.12.2015).

13. Current Trends in Remote and Virtual Lab Engineering. Where are we in 2013? / S. Seiler et al. // International Journal of Online Engineering. – 2013. – V. 9. – № 6. pp. 12 – 16.

14. D. Jannach. Recommender systems: an introduction / Jannach, D., Zanker, M., Felfernig. – Cambridge University Press, 2011. – p. 352.

15. D. Kanev. Analysis of automated learning systems architectures and architecture development of intelligent tutoring //Proceedings of International Conference. Interactive Systems And Technologies: The Problem of Human-Computer Interaction. – Collection of scientific papers. – Ulyanovsk: 2013. – September. – pp. 152-162.

16. D. Kanev. Developing the simulator based on the example of the impulse generator of accurate amplitude g5-75 device //INTERACTIVE SYSTEMS: Problems of Human - Computer Interaction. – Collection of scientific papers. – Ulyanovsk: USTU, 2015. – pp. 153-158.

17. D. Kanev. Intelligent automated training system CAD architecture// Proceedings of International Conference. Interactive Systems And Technologies: The Problem of Human-Computer Interaction. – Collection of scientific papers. – Ulyanovsk: 2011. – September. – pp. 189-198.

18. Deshpande M., Karypis G. Item-based top-n recommendation algorithms //ACM Transactions on Information Systems (TOIS). – 2004. – V. 22. – №. 1. – pp. 143-177.

19. DreamBox Learning: Using technology to provide individualized learning [Электронный ресурс] – Режим доступа: http://www.dreambox.com/wp-content/uploads/downloads/pdf/DreamBox_Using_Technology_for_Individualized_Learning.pdf (дата обращения: 20.12.2015).

20. E. Madenci, I. Guven. The Finite Element Method and Applications in Engineering Using ANSYS / E. Madenci, I. Guven. – Springer US, 2015. – p. 657.
21. Esichaikul V., Lamnoi S., Bechter C. Student modelling in adaptive e-learning systems //Knowledge Management & E-Learning: An International Journal (KM&EL). – 2011. – V. 3. – №. 3. – pp. 342-355.
22. Felfernig A. et al. An overview of direct diagnosis and repair techniques in the WeeVis recommendation environment //25th Intl. Workshop on Principles of Diagnosis. – 2014. – pp. 1-6.
23. Felfernig A. et al. The VITA financial services sales support environment //PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE. – Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 2007. – T. 22. – №. 2. – pp. 1692-1699.
24. Felfernig A., Burke R. Constraint-based recommender systems: technologies and research issues //Proceedings of the 10th international conference on Electronic commerce. – ACM, 2008. – pp. 1–10.
25. Galeev I., Tararina L., Kolosov O. Adaptation on the basis of the skills overlay model. // Proceedings of the 4th IEEE International Conference on Advanced Learning Technologies (ICALT 2004). – Finland, Jornsuu, 2004. – pp.648-650.
26. Galeev I., Tararina L., Kolosov O., Kolosov V. Structure and implementation of partially integrated adoptive learning environment //Proceedings of E-Learn 2003. – USA, Arizona, Phoenix, 2003. – pp. 2151-2154.
27. Galeev I.K., Chepegin V.I., Sosnovsky S.A. MONAP: Models, Methods and Applications// Proceedings of the International Conference KBCS 2000. – Mumbai, India, 2000. – pp. 217-228.
28. Giannotti E., Ponta D. Hypertext and Hypermedia as learning tools in science and technology //Proceedings of the IFIP WG3. 4 Working Conference on Computer Mediated Education of Information Technology Professionals and Advanced End-Users. – North-Holland Publishing Co., 1993. – pp. 335-339.
29. Grapple personalization and adaptation in learning management systems / P. De Bra, D. Smits, K. van der Sluijs et al. //In ED-MEDIA '10: Proc. of World Conference

on Educational Multimedia, Hypermedia and Telecommunications 2010. – pp. 3029-3038.

30. Gubsky D. S., Mamay I. V., Zemlyakov V. V. Virtual Laboratory for Microwave Devices // Progress In Electromagnetics Research Symposium (PIERS). Stockholm, Sweden, Aug. 12 – 15, 2013. – pp. 527 – 530.

31. Huang M. J., Huang H. S., Chen M. Y. Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning approach //Expert Systems with Applications. – 2007. – V. 33. – №. 3. –pp. 551-564.

32. IDEA Professional [Электронный ресурс]. — Режим доступа: <http://www.linkundlink.de/cms/produkte/autorensystem-idea-professional> (дата обращения: 20.12.2014).

33. IDEF [Электронный ресурс]. — Режим доступа: <http://www.idef.com/idef0.htm> (дата обращения: 20.12.2014).

34. jQuery [Электронный ресурс]. – Режим доступа: <http://jquery.com/> (дата обращения: 20.12.2014).

35. Khrib M. K., Jemn M., Nasraoui O. Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval //Advanced Learning Technologies, 2008. ICALT'08. Eighth IEEE International Conference on. – IEEE, 2008. – pp. 241-245.

36. Latest data shows Android with 81.2% of the global smartphone market for 2014 [Электронный ресурс] – Режим доступа: http://www.phonearena.com/news/Latest-data-shows-Android-with-81.2-of-the-global-smartphone-market-for-2014_id65367 (дата обращения: 20.12.2014).

37. Lops P., De Gemmis M., Semeraro G. Content-based recommender systems: State of the art and trends //Recommender systems handbook. – Springer US, 2011. – pp. 73-105.

38. M. Kula. Metadata Embeddings for User and Item Cold-start Recommendations // In Proceedings of the 2nd Workshop on New Trends on Content-Based Recommender Systems co-located with 9th ACM Conference on Recommender Systems (RecSys 2015), september 16-20. – 2015, Vienna, Austria.

39. Mahnane L., Laskri M. T., Trigano P. A model of adaptive e-learning hypermedia system based on thinking and learning styles //International Journal of Multimedia and Ubiquitous Engineering. – 2013. – V. 8. – №. 3. – pp. 336-350.
40. Marschik P. Evaluation of collaborative filtering algorithms //Doktorska disertacija. – 2010
41. Melville P., Sindhvani V. Recommender systems //Encyclopedia of machine learning. – Springer US, 2011. – С. 829-838.
42. Mustafa Y. E. A., Sharif S. M. An approach to adaptive e-learning hypermedia system based on learning styles (AEHS-LS): Implementation and evaluation //International Journal of Library and Information Science. – 2011. – V. 3. – №. 1. – pp. 15-28.
43. MySQL. Справочник по языку: перевод с английского / компания MySQL АВ. - Москва [и др.]: Вильямс, 2005. - 429 с.
44. Numerical simulation of gas pipeline networks: theory, computational implementation, and industrial applications / Seleznev V.E., Aleshin V.V., Ilkaev R.I., et al. // Ed. by V. E. Seleznev. – Moscow: KomKniga, 2005. – 720 p.
45. Ontology Development 101: A Guide to Creating Your First Ontology [Электронный ресурс]. — Режим доступа: http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html (дата обращения: 20.11.2015).
46. Pazzani M. J., Billsus D. Content-based recommendation systems //The adaptive web. – Springer Berlin Heidelberg, 2007. – pp. 325-341.
47. РСМАТ, an adaptive learning platform / Couto P. et al. //ICIPT 2013: 8th International Conference on Information Processing, Management and Intelligent Information Technology (ICIPM, ICIP). – Advanced Institute of Convergence Information Technology (AICIT), 2013.
48. Peña-Ayala A. Intelligent and Adaptive Educational-Learning Systems //Smart innovation, systems and technologies. – 2013. – Т. 17.
49. Ploum E. L. M. Authoring of Adaptation in the GRAPPLE Project //Technische Universiteit Eindhoven. – 2009.

50. Raphael—JavaScript Library [Электронный ресурс]. —Режим доступа: <http://raphaeljs.com/> (дата обращения: 20.11.2013).

51. Sosnovsky S., Brusilovsky P. Evaluation of topic-based adaptation and student modeling in QuizGuide //User Modeling and User-Adapted Interaction. – 2015. – V. 25. – №. 4. – pp. 371-424.

52. SUGGEST: Recommendation Engine [Электронный ресурс] – Режим доступа: http://glaros.dtc.umn.edu/gkhome/suggest/overview?cm_mc_uid=58961945460614586694889&cm_mc_sid_50200000=1458678440 (дата обращения: 20.12.2015).

53. SWRL: A Semantic Web Rule Language Combining OWL and RuleML [Электронный ресурс]. — Режим доступа: <https://www.w3.org/Submission/SWRL/> (дата обращения: 20.11.2015).

54. Taylor I. J., Harrison A. From P2P to Web services and grids: peers in a client/server world. – Springer Science & Business Media, 2006.

55. The Protégé Ontology Editor and Knowledge Acquisition System [Электронный ресурс]. — Режим доступа: <http://protege.stanford.edu/> (дата обращения: 20.11.2015).

56. Weerawarana S. et al. Web services platform architecture: SOAP, WSDL, WS-policy, WS-addressing, WS-BPEL, WS-reliable messaging and more. – Prentice Hall PTR, 2005.

57. Автоматное программирование [Электронный ресурс]. — Режим доступа: http://ru.wikipedia.org/wiki/Автоматное_программирование (дата обращения: 20.11.2013).

58. Алисейчик П. А. и др. Компьютерные обучающие системы //Интеллектуальные системы. – 2004. – Т. 8. – №. 1-2. – С. 5-44.

59. Алтунин, В.К. Проектирование компьютерных систем обучения и интеллектуальной тренажерной подготовки / В.К. Алтунин, А.М Стручков; под ред. О.М. Туровского. – Тверь: ЦПС, 2004. – 204 с.

60. Афанасьев А.Н., Войт Н.Н., Канев Д.С. Математическое моделирование процесса инженерного обучения в сложных инфокоммуникационных образовательных системах // Радиотехника. 2014 № 7. С. 133-136.

61. Афанасьев А. Н., Войт Н. Н. Анализ парадигм, разработка архитектуры и компонентно-сервисной платформы обучения: модель предметной области в диаграмматике uml-языка // Вестник УлГТУ. – 2011. – № 4. – С. 29-39.

62. Афанасьев А.Н., Войт Н.Н., Гульшин В. А., Канев Д.С. Реализация тренажерных систем на базе автоматного подхода // Шестая международная конференция «Системный анализ и информационные технологии», г. Светлогорск, 15-20 июня 2015 г. – Т. 1. – С. 254-264.

63. Афанасьев А.Н., Войт Н.Н., Гульшин В.А., Канев Д. С. Разработка методов управления и диагностики обучающихся в тренажёрных системах // Труды Конгресса по интеллектуальным системам и информационным технологиям «IS&IT'15». Научное издание в 3-х томах. Таганрог: Изд-во ЮФУ, 2015. – Т. 2 – С. 3-7

64. Афанасьев А.Н., Войт Н.Н., Гульшин В.А., Канев Д.С. Опыт разработки компьютерного обучающего тренажера прибора Г5-75 // Современные проблемы проектирования, производства и эксплуатации радиотехнических систем: Сборник научных трудов девятой Всероссийской научно-практической конференции (с участием стран СНГ) г. Ульяновск, 1-2 октября 2015 г. – Ульяновск: УлГТУ, 2015. – С. 169–173.

65. Афанасьев А.Н., Войт Н.Н., Канев Д.С. Модель и метод разработки и анализа компьютерных тренажеров // Автоматизация процессов управления. – 2015. – № 2 (40). – С. 64-71

66. Афанасьев А.Н., Войт Н.Н., Канев Д.С. Разработка компьютерных моделей для организации процесса инженерного обучения в сложной инфокоммуникационной среде // Сборник научных трудов 6 Всероссийской научно-технической конференции "Информатика и вычислительная техника ИВТ-2014". - Ульяновск: УлГТУ, 2014. - С. 15-20.

67. Афанасьев А.Н., Войт Н.Н., Канев Д.С. Разработка учебно-методического наполнения модели предметной области интеллектуальной обучающей систем на примере САПР КОМПАС // Информатика и вычислительная техника: сборник научных трудов /под ред. В. Н. Негоды. - Ульяновск: УлГТУ, 2010. - С. 20-26.

68. Афанасьев А.Н., Войт Н.Н., Канев Д.С., Гульшин, В.А. Моделирование виртуального тренажера на основе автоматного подхода // Радиотехника. – 2015. – № 6. – С. 55-58.

69. Афанасьев, А.Н. Разработка математического обеспечения для организации персонифицированного инженерного обучения в сложной инфо-коммуникационной образовательной среде / А.Н. Афанасьев, Н.Н. Войт, Д.С. Канев, Е.Ю. Воеводин // Труды Конгресса по интеллектуальным системам и информационным технологиям «IS&IT'14». Научное издание в 4-х томах. –М.: Физматлит, 2014. – Т. 2. – С. 203 – 210.

70. Ахремчик, О.Л. Тренажерные комплексы как составляющие инновационной деятельности в области обучения проектированию систем автоматизации / О.Л. Ахремчик // Инновации в науке и образовании 2007: тр. науч. конф. – Калининград, 2007. – С. 82 – 83.

71. Башмаков А.И., Башмаков И.А. Разработка компьютерных учебников и обучающих систем. - М.: Информационно-издательский дом "Филинь", 2003, - 616 с.

72. Башмаков И.А., Башмаков И.А. Интеллектуальные информационные технологии. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2006. — 302 с.

73. Брейкин Е. А. Рекомендательная система на основе коллаборативной фильтрации // Молодой ученый. — 2015. — №13. — С. 31-33.

74. Брусиловский, П. Л. Адаптивные обучающие системы в World Wide Web: обзор имеющихся в распоряжении технологий [Электронный ресурс] / П. Л. Брусиловский //International Forum of Educational Technology & Society – Режим доступа: <http://ifets.ieee.org/russian/depository/WWWITS.html> (дата обращения: 20.11.2013).

75. Васильев А. Н. Java. Объектно-ориентированное программирование: учебное пособие для магистров и бакалавров: базовый курс / Васильев А. Н. - Санкт-Петербург [и др.]: Питер, 2012. — 396 с.

76. Войт Н. Н. Разработка графического инструментария для построения учебного сценария // труды международной конференции «Континуальные алгебраические логики, исчисления и нейроинформатика в науке и технике». — № 2. — Ульяновск: УлГТУ:2006. - С. 18.

77. Войт Н.Н., Канев Д.С. Графический конструктор модели предметной области САПР // Информатика и вычислительная техника: сборник научных трудов. - Ульяновск: УлГТУ, 2010. - С. 134-141.

78. Войт Н.Н., Канев Д.С. Реализация клиент-серверной компьютерной системы обучения САПР // Информатика и вычислительная техника: сборник научных трудов / под ред. Н. Н. Войта. - Ульяновск: УлГТУ, 2011. - С. 162-165.

79. Войт Н.Н., Канев Д.С. Реализация компонента «файл-менеджер» для трансфера файлов в интеллектуальной системе обучения САПР // Информатика, моделирование, автоматизация проектирования: сборник научных трудов / под ред. Н. Н. Войта. – Ульяновск: УлГТУ, 2011. – С. 83-90.

80. Войт, Николай Николаевич. Разработка методов и средств адаптивного управления процессом обучения в автоматизированном проектировании: диссертация ... кандидата технических наук: 05.13.12 / Войт Николай Николаевич; [Место защиты: Ульян. гос. техн. ун-т]. - Ульяновск, 2009. - 234 с.

81. Гаврилова, Т. А. Базы знаний интеллектуальных систем: учебное пособие для вузов / Гаврилова, Т. А., Хорошевский, В. Ф.; - Санкт-Петербург: Питер, 2000. — 383 с.

82. Герберт Шилдт. Полный справочник по Java SE 6. — 7-е изд. — М.: «Вильямс», 2007. 1040 с.

83. Гладков Л. А., Курейчик В. В., Курейчик В. М. Генетические алгоритмы: Учебное пособие. — 2-е изд. — М: Физматлит, 2006. —320 с.

84. Гладун А. Я., Рогушина Ю. В. Онтологии в корпоративных системах //Корпоративные системы. – 2006. – №. 1. – С. 41-47.

85. Гомзин А. Г., Коршунов А. В. Системы рекомендаций: обзор современных подходов //Труды Института системного программирования РАН. – 2012. – Т. 22.
86. Д. Флэнаган, Ю. Мацумото. Язык программирования Ruby/ пер. с англ. Н. Вильчинский. — 1-е изд. — СПб.: Питер, 2011. — 496 с.
87. Джарратано Д., Райли Г. Экспертные системы: принципы разработки и программирование. М.: Вильямс, 2007. — 1152 с.
88. Дозорцев В. М. Компьютерные тренажеры для обучения операторов технологических процессов. — М.: СИНТЕГ, 2009. — 372 с.
89. Дозорцев, В.М. Компьютерные тренажеры для обучения операторов технологических процессов / В.М. Дозорцев. – М.: СИНТЕГ, 2009. – 372 с.
90. Доррер А. Г., Иванилова Т. Н. Моделирование интерактивного адаптивного обучающего курса //Современные проблемы науки и образования. – 2007. – №. 5. – С. 52-59.
91. Дьяконов В. Internet. Настольная книга пользователя. – Litres, 2015. — 577 с.
92. Ершов Ю. Л., Палютин Е. А. Математическая логика: Учебное пособие. СПб.: Издательство «Лань», 2004. — 336 с.
93. Заботина Н. Н. Проектирование информационных систем: Учебное пособие //М.: НИЦ ИНФРА-М. – 2014. — 336 с.
94. Канев Д. С. Исследование моделей предметных областей в АОС САПР// информатика, моделирование, автоматизация проектирования: сборник научных трудов / под ред. Н. Н. Войта. - Ульяновск: УлГТУ, 2010. - С. 255-263.
95. Канев Д.С. Анализ моделей представления предметной области в системах автоматизированного обучения // Информатика и вычислительная техника: сборник научных трудов / под ред. Н. Н. Войта. - Ульяновск: УлГТУ, 2012. - С. 270-285.
96. Канев Д.С. Обзор Javascript библиотек для редактирования диаграмм // Информатика и вычислительная техника: сборник научных трудов 5-й Всероссийской научно-технической конференции аспирантов, студентов и

молодых ученых ИВТ-2013 / под ред. Н. Н. Войта. – Ульяновск: УлГТУ, 2013. - С. 85-94.

97. Канев Д.С. Разработка онтологической модели предметной области и алгоритма формирования сценария для интеллектуальной системы обучения// Информатика, моделирование, автоматизация проектирования: сборник научных трудов / под ред. Н. Н. Войта. – Ульяновск: УлГТУ, 2012. – С. 214-218

98. Канев Д.С. Разработка онтологической модели предметной области машиностроительного проектирования и её реализация для интеллектуальной автоматизированной обучающей системы: диссертация ... магистра техники и технологии. Ульяновский государственный технический университет, Ульяновск, 2012.

99. Канев Д.С. Сравнение RMI- и SOAP-технологий // Информатика и вычислительная техника: сборник научных трудов / под ред. Н. Н. Войта. – Ульяновск: УлГТУ, 2011. – С. 263-264.

100. Канев, Д. С. Разработка обобщенной структуры тренажерного тренажера // Информатика, моделирование, автоматизация проектирования: сб. науч. тр. 5-й Всерос. шк. - семинара аспирантов, студентов и молодых ученых: ИМАП-2013, 22-23 окт. - Ульяновск: УлГТУ, 2013. - С. 84-90.

101. Канев, Д.С. Разработка модели предметной области для организации процесса инженерного обучения в сложных инфо-коммуникационных образовательных системах / А.Н. Афанасьев, Н.Н. Войт, Д.С. Канев // Труды XXI Всероссийской научно-методической конференции Телематика' 2014. – 2014. – Том 1. Секция А. – С. 262 – 263. Участие 30%

102. Карпенко А.П., Добряков А.А. Модельное обеспечение автоматизированных обучающих систем. Обзор. // Наука и образование. — 2011. — № 7. [Электронный ресурс]. URL: <http://technomag.edu.ru/doc/193116.html> (дата обращения: 15.07.2013).

103. Касьянов В.Н, Касьянова Е.В. Адаптивные системы и методы дистанционного обучения // Информационные технологии в высшем образовании. — 2004. — Т.1, N 4. — С. 40–60.

104. Касьянова Е.В. WARE - адаптивная система поддержки дистанционного обучения программированию // Труды Международной конференции. «Вычислительные и информационные технологии в науке и образовании». — Павлодар, 2006. — Том 1. — С. 606-615.

105. Кириллов В. В. Введение в реляционные базы данных. – БХВ-Петербург, 2012. — 464 с.

106. Климов, Александр Петрович. JavaScript на примерах / Александр Климов. - 2-е изд. - Санкт-Петербург: БХВ-Петербург, 2009. - 323 с.

107. Ключевые термины ФГОС [Электронный ресурс] – Режим доступа: <http://www.ulsu.ru/com/schools/opalihin/hfurj/SDVGFDR/> (дата обращения: 20.12.2015).

108. КОМПАС-3D V16. Руководство пользователя [Электронный ресурс]. — [Б.м]: Аскон, 2015.

109. КОМПАС-МАСТЕР [Электронный ресурс]. — [Б.м]: Аскон, 2015.

110. Краснянский М. Н., Карпушкин С. В., Дедов Д. Л. Системный подход к проектированию автоматизированной информационной системы обучения студентов и тренинга операторов химико-технологических систем // Вестник ТГТУ. – 2009. – №4. – С. 926-935.

111. Красовский А.А. Основы теории авиационных тренажеров. – М.: Машиностроение, 1995. - 304 с.

112. Кристиан Нейгел и др. C# 5.0 и платформа .NET 4.5 для профессионалов. — М.: «Диалектика», 2013. — 1440 с.

113. Кудрявцев В.Б., Алисейчик П.А., Вашик К. Моделирование процесса обучения // Фундаментальная и прикладная математика. 2009. Т. 15. № 05. С. 111-169.

114. Кузнецов О. П., Суховеров В. С., Шипилина Л. Б. Онтология как систематизация научных знаний: структура, семантика, задачи //Технические и программные средства систем управления, контроля и измерения: Труды II Росс. конф. с международным участием. – 2010. – С. 762-773.

115. Магид, С.И. Проблемы современного энергетического тренажеростроения через призму терминологии / С.И. Магид, И.Ш. Загретдинов, С.В. Мищеряков и др. // Энергосбережение и водоподготовка. – 2007. – № 1. – С. 43 – 51.

116. Матлин А.О., Фоменков С.А. Построение автоматизированной системы создания интерактивных тренажеров // Известия Волгоградского государственного технического университета. Сер. Актуальные проблемы управления, вычислительной техники и информатики в технических системах. – 2010 – Вып.9 – №11(71). – С. 57-59.

117. Матус П., Чуйко М. Динамический программный тренажер // Наука и инновации. – 2009. – № 4. – С. 55-56.

118. Михелькевич, В.Н. Математическая подготовка специалистов по связям с общественностью: компетентностный подход / В.Н. Михелькевич, С.Г. Афанасьева. – Самара: СамГТУ, 2008. – 160 с.

119. Найханова Л. В. Основные типы семантических отношений между терминами предметной области // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2008. – №. 1.

120. Норенков И. П. Основы автоматизированного проектирования / И. П. Норенков. – Москва: Изд-во МГТУ им. Н.Э. Баумана, 2009. – 336 с.

121. Ньюкомер Э. ВЕБ-сервисы XML, WSDL, SOAP и UDDI. – М: Питер, 2003. - 253 с.

122. ОБ УТВЕРЖДЕНИИ И ВВЕДЕНИИ В ДЕЙСТВИЕ ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ ПО НАПРАВЛЕНИЮ ПОДГОТОВКИ 151900 КОНСТРУКТОРСКО-ТЕХНОЛОГИЧЕСКОЕ ОБЕСПЕЧЕНИЕ МАШИНОСТРОИТЕЛЬНЫХ ПРОИЗВОДСТВ (КВАЛИФИКАЦИЯ (СТЕПЕНЬ) "БАКАЛАВР") [Электронный ресурс]: Приказ Министерства образования и науки Российской Федерации от 24 декабря 2009 г. N 827 (ред. от 31.05.2011 N 1975). – Режим доступа: Система Гарант

123. Об утверждении Квалификационного справочника должностей руководителей, специалистов и других служащих [Электронный ресурс]: Постановление Минтруда РФ от 21 августа 1998 г. N 37 (ред. от 12 февраля 2014 г. № 96). – Режим доступа: Система Гарант.

124. Об утверждении профессионального стандарта "Специалист по компьютерному конструированию" [Электронный ресурс]: приказ Министерства труда и социальной защиты РФ (по состоянию на 22.06.2015). – Режим доступа: Система Гарант.

125. Онтологии и тезаурусы: модели, инструменты, приложения: учебное пособие / Добров Б. В., Иванов В. В., Лукашевич Н. В. и др. - Москва: Бинوم. Лаборатория знаний: Интернет-Университет Информационных Технологий, 2009. - (Основы информационных технологий). - 172 с.

126. Прототипное программирование [Электронный ресурс]. Режим доступа: http://ru.wikipedia.org/wiki/Прототипно-ориентированное_программирование (дата обращения: 20.11.2013).

127. Рыбина Г. В. Обучающие интегрированные экспертные системы: некоторые итоги и перспективы // Искусственный интеллект и принятие решений. - 2008. - № 1. С.22 - 46.

128. Сафаров Р. Х. Состояние и пути развития компьютерного тестирования в профессиональном образовании //Филология и культура. – 2010. – №. 22.

129. Свидетельство № 2010613515 Российская Федерация. Графический конструктор модели предметной области САПР: свидетельство о государственной регистрации программы для ЭВМ / Афанасьев А.Н., Войт Н.Н., Канев Д.С.; заявитель и правообладатель Ульянов. гос. техн. ун-т. – № 2010611999; заявл. 13.04.2010; зарегистр. 28.05.2010.

130. Свидетельство № 2013610083 Российская Федерация. Интеллектуальный конструктор тестов: свидетельство о государственной регистрации программы для ЭВМ / Афанасьев А.Н., Войт Н.Н., Канев Д.С., Беляева М.А.; заявитель и правообладатель Ульянов. гос. техн. ун-т. – № 2012619698; заявл. 09.10.2012; зарегистр. 09.01.2013. Афанасьев А.Н., Войт Н. Н, Канев Д. С.

131. Свидетельство № 2014661213 Российская Федерация. Графический редактор онтологической модели предметной области САПР: свидетельство о государственной регистрации программы для ЭВМ / Афанасьев А.Н., Войт Н.Н., Канев Д.С.; заявитель и правообладатель Ульян. гос. техн. ун-т. – № 2014619103; заявл. 10.09.2014; зарегистр. 27.10.2014.

132. Свидетельство № 2015616393 Российская Федерация. Автоматическое построение модели предметной области САПР на основе детали из КОМПАС-3D: свидетельство о государственной регистрации программы для ЭВМ / Афанасьев А.Н., Войт Н.Н., Канев Д.С.; заявитель и правообладатель Ульян. гос. техн. ун-т. – № 2015613505; заявл. 10.09.2014; зарегистр. 27.10.2014.

133. Свидетельство № 2015616397 Российская Федерация. Формирование индивидуального сценария обучения на основе динамических механизмов взаимодействия моделей пространства обучения: свидетельство о государственной регистрации программы для ЭВМ / Афанасьев А.Н., Войт Н.Н., Канев Д.С.; заявитель и правообладатель Ульян. гос. техн. ун-т. – № 2015613506; заявл. 24.04.2015; зарегистр. 09.06.2015.

134. Свидетельство № 2015617292 Российская Федерация. Веб-ориентированная система представления учебно-методического материала: свидетельство о государственной регистрации программы для ЭВМ / Афанасьев А.Н., Войт Н.Н., Канев Д.С.; заявитель и правообладатель Ульян. гос. техн. ун-т. – № 2015613947; заявл. 12.05.2015; зарегистр. 06.07.2015.

135. Свидетельство № 2015617322 Российская Федерация. Платформа для автоматизированной обучающей системы: свидетельство о государственной регистрации программы для ЭВМ / Афанасьев А.Н., Войт Н.Н., Канев Д.С.; заявитель и правообладатель Ульян. гос. техн. ун-т. – № 2015613908; заявл. 12.05.2015; зарегистр. 07.07.2015.

136. Сергеев С.Ф. Виртуальные тренажеры: проблемы теории и методологии проектирования // Биотехносфера. – 2010. – № 2(8). – С. 15-20.

137. Стандарт профессиональной деятельности инженера-проектировщика [Электронный ресурс]. – Режим доступа: <http://npirf.ru/wp->

content/uploads/2015/08/03.-

%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82-
%D0%BF%D1%80%D0%BE%D1%84.-%D0%B4%D0%B5%D1%8F%D1%82.-
%D0%B8%D0%BD%D0%B6%D0%B5%D0%BD%D0%B5%D1%80%D0%B0-
%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%
BE%D0%B2%D1%89%D0%B8%D0%BA%D0%B0-28-10-2015.pdf (Дата
обращения: 21.02.2014).

138. Техническое описание "Инженерная графика CAD" [Электронный ресурс]. – Режим доступа: http://www.mami.ru/storage/files/ws/cad/TehOpisanie_05_CAD_Versiya_10.2015.doc. – (Дата обращения: 21.02.2014).

139. Трехмерное моделирование деталей и сборочных единиц в системе КОМПАС-3D [Электронный ресурс]. – Режим доступа: <http://certification.ascon.ru/source/documents/certification/2015/M2-instrukcia.pdf>. – (Дата обращения: 21.02.2014).

140. Трухин А. В. Анализ существующих в РФ тренажерно-обучающих систем // Открытое и дистанционное образование, 2008. – № 1. – С. 32-39.

141. Трухин А.В. Автоматизированная тренажерно-обучающая система: компьютерный тренажер и язык описания сценариев // Открытое и дистанционное образование. – 2007. – № 3 (27). – С. 47-56.

142. Фионова, Л.Р. Адаптивное управление в системе непрерывного образования на основе компетентностного подхода (на примере сферы документационного обеспечения управления): автореф. дис. ... д-ра. техн. наук: 05.13.10 / Фионова Людмила Римовна. – Пенза, 2009. – 32 с.

143. Холзнер, Стивен. XML: перевод с английского / Холзнер С. - 2-е изд. - Санкт-Петербург [и др.]: Питер, 2004. - (Энциклопедия. Наиболее полное и подробное руководство). - 1100 с.

144. Черепашков, А.А. Комплекс компьютерных инженерных тренажеров для обучения конструированию / А.А. Черепашков // Компьютерные технологии обучения: концепции, опыт, проблемы. – Самара: СамГТУ, 1997. – С. 23.

145. Черепашков, А.А. Тренажер для конструктора по устойчивости / А.А. Черепашков // Проблемы фундаментализации инженерного образования в СГАУ. – Самара, 1994. – С. 44 – 45.

146. Черепашков, А.А. Тренажер для подготовки к автоматизированному проектированию структур силовых конструкций: дис. ... канд. техн. наук: 05.13.12 / Черепашков Андрей Александрович. – Куйбышев: КуАИ, 1987. – 199 с.

147. Черепашков, Андрей Александрович. Методы и средства обучения автоматизированному проектированию в машиностроении: диссертация ... доктора технических наук: 05.13.12 / Черепашков Андрей Александрович; [Место защиты: Сам. гос. аэрокосм. ун-т им. С.П. Королева]. - Самара, 2014. - 434 с

148. Шабалина О.А. Модель процесса обучения и ее интерпретация в обучающей компьютерной игре / О.А Шабалина // Вестник СГТУ. – 2013. – № 2(70). Выпуск 1. С.158-167.

149. Шабалина, О.А. Модели и методы адаптации для управления знаниями с помощью адаптивных обучающих систем / О. А. Шабалина // Открытое образование: журнал. Приложение. Открытое образование и информационные технологии. - 2005. - Приложение. - С. 306-310.

150. Шаров О.Г., Афанасьев А.Н. Автоматная графическая грамматика // Вестник Ульяновского государственного технического университета. 2005. № 1 (29). С. 54-56.

151. Юрков Н.К. Интеллектуальные компьютерные обучающие системы. Монография. // Пенза: Изд-во Пенз. гос. ун-та, 2010. - 304 с.

Приложение 1. Список знаний, умений и навыков для компетенции «владеет навыками твердотельного моделирования»

Проектировщик должен знать:

- общие сведения о телах;
- начало построения модели;
- элемент выдавливания;
- выполнение операции «выдавливание»;
- выполнение операции «вырезать выдавливанием»;
- зависимость результата операции выдавливания от сечения;
- указание сечения элемента выдавливания;
- требования к эскизу элемента выдавливания;
- направляющий объект для выдавливания;
- выбор направления выдавливания;
- глубина выдавливания;
- выдавливание на заданное расстояние;
- выдавливание через все;
- выдавливание до вершины;
- выдавливание до поверхности;
- выдавливание до ближайшей поверхности;
- задание расстояния до объекта;
- угол уклона;
- тонкая стенка;
- результат операции;
- результат добавления материала;
- результат удаления материала;
- элемент вращения;
- выполнение операции «вырезать вращением»;
- выполнение операции «вращение»;
- сечение элемента вращения;
- зависимость результата операции вращения от сечения;
- указание сечения элемента вращения;
- требования к эскизу элемента вращения;
- ось вращения;
- типы элемента вращения;
- направление вращения;
- угол вращения;
- вращение на заданный угол;
- вращение до вершины;
- вращение до поверхности;
- кинематический элемент;

- выполнение кинематической операции;
- сечение кинематического элемента;
- требования к эскизу-сечению;
- траектория кинематического элемента;
- требования к траектории;
- тип движения сечения;
- элемент по сечениям;
- выполнение операции «по сечениям»;
- выполнение операции «вырезать по сечениям»;
- сечения элемента по сечениям;
- требования к эскизам-сечениям;
- осевая линия элемента по сечениям;
- требования к осевой линии;
- способ построения элемента у крайних сечений;
- траектория соединения сечений;
- способ генерации траектории;
- замкнутая или разомкнутая траектория;
- придание толщины граням тела или поверхности;
- булева операция над телами;
- отображение булевой операции в дереве построения;
- операции редактирования тел;
- фаска;
- скругление;
- скругление с постоянным радиусом;
- скругление с переменным радиусом;
- сглаживание и обход углов;
- скругление с сохранением кромки или с сохранением поверхности;
- остановка скругления;
- преобразование тел в оболочку;
- ребро жесткости;
- требования к эскизу ребра жесткости;
- формирование ребра жесткости;
- положение ребра;
- уклон граней ребра;
- уклон граней;
- отсечение части модели;
- сечение поверхностью;
- сечение по эскизу;
- масштабирование тел и поверхностей;
- изменение положения тела или поверхности;
- круглое отверстие;

- простое отверстие;
- отверстие с зенковкой;
- отверстие с цековкой;
- отверстие с зенковкой и цековкой;
- коническое отверстие;
- задание глубины отверстия;
- параметры отверстия с зенковкой;
- параметры отверстия с цековкой;
- параметры конического отверстия;
- создание резьбового отверстия;
- задание формы торца отверстия;
- размещение отверстия на поверхности;
- создание отверстия по образцу;
- отверстие из библиотеки;
- область применения операции;
- общий порядок задания области применения операции;
- область применения «тела»;
- задание области применения «тела» для операций добавления материала;
- задание области применения «тела» для операций удаления материала;
- способы определения области применения «тела»;
- область применения «компоненты»;
- задание области применения «компоненты»;
- формирование области применения «компоненты» вручную;
- тела, состоящие из отдельных частей;
- набор частей: выбор частей, которые следует оставить;
- особенности изменения количества частей тела;
- многотельное моделирование;
- примеры использования многотельного моделирования;
- отображение многотельной модели в дереве построения.

Навыки и умения:

- указание элемента выдавливания;
- выполнение операции «выдавливание»;
- выполнение операции «вырезать выдавливанием»;
- указание сечения элемента выдавливания;
- определение направляющего объекта для выдавливания;
- выбор направления выдавливания;
- указание глубины выдавливания;
- выдавливание на заданное расстояние;
- выдавливание через все;
- выдавливание до вершины;
- выдавливание до поверхности;
- выдавливание до ближайшей поверхности;

- задание расстояния до объекта;
- установка угла уклона;
- задание тонкой стенки;
- добавление материала;
- удаление материала;
- выбор элемента вращения;
- выполнение операции «вырезать вращением»;
- выполнение операции «вращение»;
- указание сечения элемента вращения;
- выбор оси вращения;
- определение типа элемента вращения;
- направление вращения;
- установка угла вращения;
- вращение на заданный угол;
- вращение до вершины;
- вращение до поверхности;
- выбор кинематического элемент;
- выполнение кинематической операции;
- сечение кинематического элемента;
- установка траектории кинематического элемента;
- выбор типа движения сечения;
- выполнение операции «по сечениям»;
- выполнение операции «вырезать по сечениям»;
- сечения элемента по сечениям;
- выбор осевой линии элемента по сечениям;
- построение элемента у крайних сечений;
- определение траектории соединения сечений;
- указание способа генерации траектории;
- выбор замкнутой или разомкнутой траектории;
- придание толщины граням тела или поверхности;
- выполнение булевых операций над телами;
- отображение булевой операции в дереве построения;
- выполнение операции «фаска»;
- скругление;
- скругление с постоянным радиусом;
- скругление с переменным радиусом;
- сглаживание и обход углов;
- скругление с сохранением кромки или с сохранением поверхности;
- остановка скругления;
- преобразование тел в оболочку;
- выбор ребра жесткости;

- формирование ребра жесткости;
- установка положения ребра;
- уклон граней ребра;
- уклон граней;
- отсечение части модели;
- сечение поверхностью;
- сечение по эскизу;
- масштабирование тел и поверхностей;
- изменение положения тела или поверхности;
- построение круглого отверстия;
- построение простого отверстия;
- построение отверстия с зенковкой;
- построение отверстия с цековкой;
- построение отверстия с зенковкой и цековкой;
- построение конического отверстия;
- задание глубины отверстия;
- установка параметров отверстия с зенковкой;
- установка параметров отверстия с цековкой;
- установка параметров конического отверстия;
- создание резьбового отверстия;
- задание формы торца отверстия;
- размещение отверстия на поверхности;
- создание отверстия по образцу;
- построение отверстия из библиотеки;
- задание области применения «тела» для операций добавления материала;
- задание области применения «тела» для операций удаления материала;
- выбор способа определения области применения «тела»;
- задание области применения «компоненты»;
- формирование области применения «компоненты» вручную;

Приложение 2. Перечень рекомендаций

Примеры рекомендаций.

1. Не следует задавать параметр «Направление первой стороны» для операции фаски, если её стороны равны, так как результат построения не будет зависеть от данного параметра.

2. При построении фаски для множества рёбер с одинаковыми параметрами операции, постарайтесь выбрать как можно большее количество ребер.

3. При построении скругления для множества рёбер с одинаковыми параметрами операции, постарайтесь выбрать как можно большее количество ребер.

4. При построении операции «Уклон» для множества граней с одинаковыми параметрами операции, постарайтесь выбрать как можно большее количество граней.

5. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для четырёх и более объектов воспользуйтесь опцией «Запомнить состояние».

6. При построении параметрического ограничения «равенство длин» для четырёх и более объектов воспользуйтесь опцией «Запомнить состояние».

7. Построенные тела являются идентичными и располагаются в узлах параллелограммной сетки, воспользуйтесь операцией «Массив по сетке».

8. Построенные тела являются идентичными и располагаются в узлах концентрической сетки, воспользуйтесь операцией «Массив по концентрической сетке».

9. Построенные тела являются идентичными и располагаются вдоль кривой, воспользуйтесь операцией «Массив вдоль кривой».

10. Построенные тела являются идентичными и располагаются на точечных объектах, воспользуйтесь операцией «Массив по точкам».

11. Команды запуска операций, которые могут быть выполнены на основе созданного эскиза, можно вызвать прямо из режима эскиза. Построив в эскизе сечение тела выдавливания, можно сразу вызвать команду Операция

выдавливания. Режим редактирования эскиза автоматически завершится, будет запущена команда построения тела выдавливания.

12. Построив в эскизе сечение тела выдавливания, можно сразу вызвать команду «Вырезать выдавливанием». Режим редактирования эскиза автоматически завершится, будет запущена команда «Вырезать выдавливанием».

13. Построив в эскизе сечение тела выдавливания, можно сразу вызвать команду «Вращение». Режим редактирования эскиза автоматически завершится, будет запущена команда «Вращение».

14. Построив в эскизе сечение тела выдавливания, можно сразу вызвать команду «Вырезать вращением». Режим редактирования эскиза автоматически завершится, будет запущена команда «Вырезать вращением».

15. Сечение кинематического элемента можно указать перед запуском операции.

16. Элемент выдавливается на минимальное расстояние, при котором все точки торца оказываются за габаритным параллелепипедом модели. Воспользуйтесь способом «Выдавливание через все».

17. Глубина выдавливания определена до вершины объекта. Воспользуйтесь способом «Выдавливание до вершины».

18. Глубина выдавливания определена до грани или плоскости. Воспользуйтесь способом «Выдавливание до поверхности».

19. Построенные четыре отрезка имеют ограничения «Совпадение двух точек», «Горизонталь», «Вертикаль». Воспользуйтесь операцией «Прямоугольник».

20. Построенные четыре отрезка имеют ограничения «Совпадение двух точек». Воспользуйтесь операцией «Прямоугольник по 3 точкам».

21. Построенные отрезки имеют ограничения «Совпадение двух точек» и равны между собой. Воспользуйтесь операцией «Многоугольник».

22. Построенные отрезки имеют ограничения «Совпадение двух точек». Воспользуйтесь операцией «Непрерывный ввод объектов».

23. Построенные отрезки имеют ограничения «Совпадение двух точек» и равны между собой. Воспользуйтесь операцией «Многоугольник».

24. У построенного эллипса равные длины сторон. Воспользуйтесь операцией «Окружность».

25. На грани тела была выполнена операций «Вырезать выдавливанием» над треугольником в качестве сечения. Воспользуйтесь операцией «Фаска».

26. На грани тела была выполнена операций «Вырезать выдавливанием» над дугой и двумя отрезками в качестве сечения. Воспользуйтесь операцией «Скругление».

27. На грани тела была выполнена операций «Выдавливанием» над треугольником в качестве сечения. Воспользуйтесь операцией «Уклон граней».

Приложение 3. Описание контроллеров для виртуальных компонент системы формирования профиля компетенций проектировщика

Описание внешних контроллеров

Button (Кнопка)

Контроллер «Кнопка» подписывается на событие «action», которое происходит при взаимодействии с данным контроллером. Параметром кнопки является фикция смена состояние тренажёра. Перед сменой состояния индуцирует событие «button_click» (клик по кнопке). После смены состояния кнопка отсылает событие «state_change» (изменение состояния тренажёра) и событие «button_post» (клик по кнопке после смены состояния).

ButtonBox (Кнопка со светодиодом)

Ведёт себя как контроллер «Кнопка». Дополнительно подписывается на событие «state_change» (изменение состояния тренажёра) и имеет параметр «on». При возникновении события в зависимости от состояния тренажёра включает внутренний светодиод. Является гибридом кнопки и индикатора-светодиода.

ButtonBoxImage (Кнопка с картинкой)

Ведёт себя как контроллер «Кнопка со светодиодом», но не включает светодиод, а полностью меняет свой внешний вид. Имеет дополнительные параметры «image_on» (картинка при включенном состоянии) и «image_off» (Картинка при выключенном состоянии).

ButtonFlag (Кнопка - флаг)

Ведёт себя как контроллер «Кнопка со светодиодом», но вместо включения светодиода, включает полную подсветку кнопки.

IndicatorDiode (Индикатор-светодиод)

Контроллер «Индикатор-светодиод» подписывается через ядро на событие «state_change» (изменение состояния тренажёра), при возникновении данного события контроллер в соответствии со своими параметрами определяет, должен ли сейчас он гореть или нет. Параметром индикатора является функция, возвращающая единицу, когда он должен гореть, и ноль – когда не должен.

IndicatorDiodeOpacity (Индикатор-светодиод с прозрачностью)

Ведёт себя как контроллер «Индикатор-светодиод», только индикатор загорается с прозрачностью 40%.

IndicatorNumber (Индикатор-число)

Контроллер подписывается через ядро на событие «state_change» (изменение состояния тренажёра), при возникновении данного события контроллер выводит число с форматированием, возвращаемое функцией заданной параметром. Параметрами являются: функция возвращающее число для вывода, функция вывода (выводить число или нет), количество знаков в целой части, количество знаков в дробной части.

IndicatorText (Индикатор-текст)

Ведёт себя как контроллер «Индикатор-число», только выводит не число, а текст. Позволяется задать цвет и размеры шрифта.

Reostat (Ползунок)

Контроллер ползунок представляет собой ползунок в форме круга. При его вращении изменяет состояние тренажёра и отправляет событие «state_change» (изменение состояния тренажёра). Имеет следующие параметры:

- функция смены состояние, принимает на вход текущее положение ползунка;
- различные опции: минимальный и максимальный угол, минимальное и максимальное значение, начальный угол, радиус, цвет и так далее.

Описание логических контроллеров

WarningManager (Менеджер ошибок)

Подписывается на событие «warning» (предупреждение). При возникновении данного события выводит сообщение, что тренажёр настроен не корректно.

FatalManager (Менеджер аварийных состояний)

Подписывается на событие «fatal» (аварийное состояние). При возникновении данного события выводит сообщение, что тренажёр сломался. После чего откатывает состояние.

Menu (Меню)

Выводит меню тренажёра, обрабатывает смену режима работы тренажёра. Непосредственно обрабатывает режим справки.

Oscilloscope (Осциллограф)

Подписан на событие «signal» (Сигнал). При инициировании данного события выводит на осциллограмму значение сигнала. Параметром является DOM (Document Object Model) - контейнер, в который выводится осциллограмма.

Tutor (Учитель)

Обеспечивает систему обучения работы с тренажёром. Обрабатывает режимы обучения и контроля.

Recomend (Система рекомендаций)

Контроллер подписан на событие «state_change» (изменение состояния тренажёра). Ведёт трассу пользовательских действий, проводит анализ целей и путей их достижений, формирует рекомендации.

Приложение 4. Анализ сборки «Насос»

Деталь «Вакуумная камера».

Получена применением 111 операций, из них 9 – формообразующих, 71 – построение геометрических объектов, 6 – построение эскизов, 24 – наложение параметризации. Общее количество совершённых действий при построении – 487.

Сформированы следующие рекомендации.

1. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 4 геометрических объектов эскиза «Эскиз:1» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 25%.

2. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 6 геометрических объектов эскиза «Эскиз:1» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 38%.

При выполнении рекомендаций общее количество действий уменьшится с 487 до 480 или на 1%.

Деталь «Держатель клапана».

Получена применением 13 операций, из них 1 – формообразующая, 8 – построение геометрических объектов, 2 – построение эскизов, 1 – наложение параметризации. Общее количество совершённых действий при построении – 52.

Рекомендаций для данной детали не выявлено.

Деталь «Иллюминатор».

Получена применением 23 операций, из них 3 – формообразующие, 11 – построение геометрических объектов, 3 – построение эскизов, 5 – наложение параметризации. Общее количество совершённых действий при построении – 101.

Рекомендаций для данной детали не выявлено.

Деталь «Клапан».

Получена применением 12 операций, из них 1 – формообразующая, 8 – построение геометрических объектов, 2 – построение эскизов, 0 – наложение параметризации. Общее количество совершённых действий при построении – 47.

Рекомендаций для данной детали не выявлено.

Деталь «Колесо ременной передачи».

Получена применением 81 операции, из них 6 – формообразующих, 54 – построение геометрических объектов, 7 – построение эскизов, 13 – наложение параметризации. Общее количество совершённых действий при построении – 330.

Сформирована следующая рекомендация.

1. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 4 геометрических объектов эскиза «Эскиз:6» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 25%.

При выполнении рекомендаций общее количество действий уменьшится с 330 до 328 или на 1%.

Деталь «Корпус насоса».

Получена применением 249 операций, из них 12 – формообразующих, 148 – построение геометрических объектов, 8 – построение эскизов, 80 – наложение параметризации. Общее количество совершённых действий при построении – 1044.

Сформированы следующие рекомендации.

1. У вас 2 одинаковых операции («Скругление:3», «Скругление:4»). При построении скругления для множества рёбер с одинаковыми параметрами операции, постарайтесь выбрать как можно большее количество рёбер, это уменьшит количество действий на 38%.

2. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 8 геометрических объектов эскиза «Эскиз:11» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 47%.

3. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 6 геометрических объектов эскиза «Эскиз:12» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 38%.

4. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 4 геометрических объектов эскиза «Эскиз:13» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 25%.

При выполнении рекомендаций общее количество действий уменьшится с 1044 до 1025 или на 2%.

Деталь «Крышка вакуумной камеры».

Получена применением 44 операций, из них 6 – формообразующих, 22 – построение геометрических объектов, 5 – построение эскизов, 10 – наложение параметризации. Общее количество совершённых действий при построении – 195.

Сформирована следующая рекомендация.

1. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 6 геометрических объектов эскиза «Эскиз:5» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 38%.

При выполнении рекомендаций общее количество действий уменьшится с 195 до 190 или на 3%.

Деталь «Крышка верхняя».

Получена применением 50 операций, из них 12 – формообразующих, 29 – построение геометрических объектов, 8 – построение эскизов, 0 – наложение параметризации. Общее количество совершённых действий при построении – 218.

Рекомендаций для данной детали не выявлено.

Деталь «Крышка подшипника».

Состоит из 19 операций, из них 1 – формообразующая, 10 – построение геометрических объектов, 1 – построение эскизов, 6 – наложение параметризации. Общее количество совершённых действий при построении – 80.

Сформирована следующая рекомендация.

1. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 4 геометрических объектов эскиза «Эскиз:1»

воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 25%.

При выполнении рекомендаций общее количество действий уменьшится с 80 до 78 или на 3%.

Деталь «Лопасть».

Получена применением операций, из них 2 – формообразующие, 8 – построение геометрических объектов, 2 – построение эскизов, 0 – наложение параметризации. Общее количество совершённых действий при построении – 59.

Рекомендаций для данной детали не выявлено.

Деталь «Передняя крышка».

Получена применением 127 операций, из них 9 – формообразующих, 82 – построение геометрических объектов, 6 – построение эскизов, 29 – наложение параметризации. Общее количество совершённых действий при построении – 548.

Сформированы следующие рекомендации.

1. У вас 2 одинаковых операции («Скругление:3», «Скругление:4»). При построении скругления для множества рёбер с одинаковыми параметрами операции, постарайтесь выбрать как можно большее количество рёбер, это уменьшит количество действий на 38%.

2. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 8 геометрических объектов эскиза «Эскиз:1» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 44%.

3. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 4 геометрических объектов эскиза «Эскиз:1» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 25%.

4. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 4 геометрических объектов эскиза «Эскиз:3» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 25%.

5. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 4 геометрических объектов эскиза «Эскиз:6» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 25%.

При выполнении рекомендаций общее количество действий уменьшится с 548 до 531 или на 3%.

Деталь «Подшипник скольжения».

Получена применением 22 операций, из них 2 – формообразующие, 10 – построение геометрических объектов, 2 – построение эскизов, 7 – наложение параметризации. Общее количество совершённых действий при построении – 96.

Сформирована следующая рекомендация.

1. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 4 геометрических объектов эскиза «Эскиз:2» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 25%.

При выполнении рекомендаций общее количество действий уменьшится с 96 до 94 или на 2%.

Деталь «Прокладка верхняя».

Получена применением 37 операций, из них 4 – формообразующие, 24 – построение геометрических объектов, 1 – построение эскизов, 7 – наложение параметризации. Общее количество совершённых действий при построении – 151.

Сформирована следующая рекомендация.

1. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 6 геометрических объектов эскиза «Эскиз:1» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 38%.

При выполнении рекомендаций общее количество действий уменьшится с 151 до 146 или на 3%.

Деталь «Прокладка передняя».

Получена применением 44 операций, из них 3 – формообразующие, 26 – построение геометрических объектов, 1 – построение эскизов, 13 – наложение параметризации. Общее количество совершённых действий при построении – 182.

Сформирована следующая рекомендация.

1. При построении параметрического ограничения «равенство радиусов двух дуг или окружностей» для 8 геометрических объектов эскиза «Эскиз:1» воспользуйтесь опцией «Запомнить состояние», это уменьшит количество действий на 44%.

При выполнении рекомендаций общее количество действий уменьшится с 182 до 174 или на 4%.

Деталь «Прокладка под выводящую трубку».

Получена применением 5 операций, из них 1 – формообразующая, 2 – построение геометрических объектов, 1 – построение эскизов, 0 – наложение параметризации. Общее количество совершённых действий при построении – 21.

Рекомендаций для данной детали не выявлено.

Деталь «Пружина».

Получена применением 7 операций, из них 2 – формообразующие, 2 – построение геометрических объектов, 2 – построение эскизов, 0 – наложение параметризации. Общее количество совершённых действий при построении – 27.

Рекомендаций для данной детали не выявлено.

Деталь «Ротор».

Получена применением 51 операции, из них 9 – формообразующих, 34 – построение геометрических объектов, 7 – построение эскизов, 0 – наложение параметризации. Общее количество совершённых действий при построении – 222.

Рекомендаций для данной детали не выявлено.

Деталь «Трубка входная».

Получена применением 37 операций, из них 2 – формообразующие, 26 – построение геометрических объектов, 2 – построение эскизов, 6 – наложение параметризации. Общее количество совершённых действий при построении – 146.

Рекомендаций для данной детали не выявлено.

Деталь «Трубка выводящая».

Получена применением 48 операций, из них 4 – формообразующие, 31 – построение геометрических объектов, 4 – построение эскизов, 8 – наложение параметризации. Общее количество совершённых действий при построении – 207.

Рекомендаций для данной детали не выявлено.

Деталь «Шпилька».

Получена применением 54 операций, из них 1 – формообразующая, 38 – построение геометрических объектов, 2 – построение эскизов, 12 – наложение параметризации. Общее количество совершённых действий при построении – 211.

Рекомендаций для данной детали не выявлено.

Деталь «Штифт».

Получена применением 22 операций, из них 1 – формообразующая, 18 – построение геометрических объектов, 2 – построение эскизов, 0 – наложение параметризации. Общее количество совершённых действий при построении – 83.

Рекомендаций для данной детали не выявлено.

Приложение 5. Акт о внедрении



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное
образовательное учреждение высшего образования
**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**
(УлГТУ)
Северный венец ул., д. 32,
г. Ульяновск, 432027, Россия
Тел.: (8422) 43-06-43; факс: (8422) 43-02-37
E-mail: rector@ulstu.ru http://www.ulstu.ru
ОКПО 02069378, ОГРН 1027301160226
ИНН/КПП 7325000052/732501001



АКТ

о внедрении в учебный процесс
результатов диссертационной работы Канева Д.С.

Результаты диссертационной работы Канева Д.С. «Разработка моделей, методов и средств формирования профиля компетенций проектировщика в процессах автоматизированного проектирования машиностроительных объектов (на примере САПР КОМПАС)», представленной на соискание ученой степени кандидата технических наук, внедрены в учебный процесс Ульяновского государственного технического университета при обучении студентов направления «Информатика и вычислительная техника» по специальностям бакалавр техники и технологии 23010063, магистр техники и технологии 23010068 (лекции, лабораторные и практические занятия, курсовое и дипломное проектирование, выпускные работы на степень бакалавра и диссертационные работы на степень магистра).

Заведующий кафедрой «Вычислительная техника»
ФГБОУ ВО
«Ульяновский государственный технический университет»
д.т.н., профессор

П.И. Соснин